Fed-Batch Process Modelling for State Estimation and Optimal Control

Kristensen, Niels Rode

Publication date:
2002

Document Version
Publisher's PDF, also known as Version of record

Fed-Batch Process Modelling for State Estimation and Optimal Control

A Stochastic Grey-Box Modelling Framework

Niels Rode Kristensen
Department of Chemical Engineering
Technical University of Denmark

ISBN 87-90142-83-7
Printed by Book Partner, Nørhaven Digital, Copenhagen, Denmark

Preface

This thesis was prepared at the Department of Chemical Engineering (KT) in collaboration with Informatics and Mathematical Modelling (IMM), both at the Technical University of Denmark (DTU), in partial fulfillment of the requirements for receiving the Ph.D. degree. The work presented in the thesis was carried out from August 1999 to December 2002 and was financed by DTU.

During the course of the work presented in the thesis, a number of people have provided their help and support, for which I am very grateful. First of all, I would like to thank my supervisors, Professor Sten Bay Jørgensen, KT, and Professor Henrik Madsen, IMM, for their input during the many fruitful discussions I have had with them. I would also like to thank Dennis Bonné, Lars Gregersen, John Bagterp Jørgensen and Frede Lei, who have all, in one way or the other, contributed to the work presented in the thesis. Also thanks to all of my other present and former colleagues, especially Lasse Engbo Christiansen, Morten Skov Hansen, Mads Thaysen and Christoffer Wenzel Tornøe for testing and providing suggestions for improvement to the software I have developed.

I would also like to acknowledge Professor Emeritus Torsten Bohlin, Royal Institute of Technology (KTH), Sweden, for the help I have received from him in the preparation of the first of the papers included in the back of the thesis.

Finally, sincere thanks to my family for their love and support and for at least trying to understand what my work is all about, and also to all of my friends for helping me to have a social life despite the many late nights at the office.

Lyngby, December 2002

Niels Rode Kristensen

Summary

The subject of this thesis is modelling of fed-batch processes for the purpose of state estimation and optimal control, the motivation being the shortcomings of present industrial approaches to fed-batch process operation with respect to achieving uniform operation and optimal productivity, and the resulting need for development of an appropriate model-based approach to automatic operation capable of achieving these goals. A number of requirements for such an approach are therefore listed, and a review of various approaches reported in literature is given along with a discussion of their merits with respect to meeting these requirements. This review indicates that it may be particularly advantageous to use an approach incorporating continuous-discrete stochastic state space models, which are models consisting of a set of stochastic differential equations describing the dynamics of the system in continuous time and a set of algebraic equations describing how measurements are obtained at discrete time instants. This is due to the fact that such models combine the strengths of first engineering principles models and data-driven models, neither of which are ideally suited in their own right. Based on continuous-discrete stochastic state space models, the main features of an overall framework for fed-batch process modelling, state estimation and optimal control are therefore first established, but since this framework incorporates modelling as well as experimental design and state estimation and optimal control, attention is restricted to the modelling part, to facilitate which a grey-box modelling framework is proposed.

This framework is based on a grey-box modelling cycle, the idea of which is to facilitate the development of models of fed-batch processes for the purpose of state estimation and optimal control. This modelling cycle, which comprises six different tasks, is the main result of the thesis, and much emphasis is put on describing methods and tools to facilitate its individual tasks. In this regard, particular emphasis is put on describing the extension of an existing parameter estimation method for continuous-discrete stochastic state space models to make it more readily applicable to models of fed-batch processes and the implementation of this method in a computer program called **CTSM**, and it is shown that this program is superior, both in terms of quality of estimates and in terms of reproducibility, to another program implementing a similar estimation method. Additional tools, implemented in MATLAB, which facilitate other important tasks within the grey-box modelling cycle are also described, and based on all of the individual tasks of the modelling cycle a grey-box modelling algorithm that facilitates systematic iterative model improvement is presented, and its key features and limitations are subsequently discussed.

A particularly important such feature is that the methodology provided by the grey-box modelling algorithm facilitates pinpointing of model deficiencies based on information extracted from experimental data and subsequently allows the structural origin of these deficiencies to be uncovered as well to provide guidelines for model improvement. This is a very powerful feature not shared by other approaches to grey-box modelling reported in literature, which rely solely on the model maker to determine how to improve the model, and it is therefore argued that, in this particular sense, the proposed methodology is more systematic, which is a key result. However, like other approaches to grey-box modelling, the proposed methodology is limited by the quality and amount of available prior physical knowledge and experimental data, and a discussion of the implications of these limitations is also given. The performance of the proposed methodology is demonstrated through a number of application examples, based on which it is then argued that, although no rigorous proof of convergence exists, the grey-box modelling algorithm may in fact converge for certain simple systems, and that, in any case, the proposed methodology can be applied to facilitate faster model development. A generalized version of the grey-box modelling algorithm, which is not limited to modelling of fed-batch processes for the purpose of state estimation and optimal control but can be applied to model a variety of systems for different purposes, is also presented.

Resumé på dansk

Emnet for denne afhandling er modellering af fed-batch processer med henblik på tilstandsestimering og optimal regulering, hvilket er motiveret af det faktum, at aktuel industriel praksis for drift af fed-batch processer ikke er i stand til at sikre et ensartet procesforløb og i særdeleshed ikke optimal produktivitet, samt af det heraf afledte behov for udvikling af en passende modelbaseret metode til automatisk drift, som er i stand til at opnå disse mål. Derfor opstilles en række krav til en sådan metode, og en række metoder fra litteraturen gennemgås med henblik på at vurdere deres evne til at opfylde disse krav. Denne gennemgang viser, at der med fordel kan benyttes en metode, som baserer sig på kontinuert-diskrete stokastiske tilstandsmodeller, dvs. modeller bestående af et sæt af stokastiske differentialligninger, der beskriver systemets dynamik i kontinuert tid, samt et sæt af algebraiske ligninger, der beskriver hvorledes der måles på systemet til diskrete tidspunkter. Dette skyldes, at sådanne modeller er i stand til at kombinere fordelene ved rent deduktive henholdsvis rent induktive modeller, hvoraf ingen i sig selv er helt ideelle. Baseret på kontinuert-diskrete stokastiske tilstandsmodeller opstilles derfor først rammerne for en overordnet metode til modellering, tilstandsestimering og optimal regulering af fed-batch processer, men da denne metode omfatter både modellering, eksperimentelt design og tilstandsestimering og optimal regulering, begrænses fokus herefter til modelleringsdelen, hvortil der foreslås en grey-box-modelleringsmetode.

Denne metode er baseret på en grey-box-modeldannelsescyklus, som kan bruges til opstilling af modeller af fed-batch processer med henblik på tilstandsestimering og optimal regulering. Denne modeldannelsescyklus, som består af seks forskellige trin, er afhandlingens hovedresultat, og der lægges vægt på at beskrive metoder og værktøjer, der kan bruges i forbindelse med hvert af disse trin. Eksempelvis lægges der særlig vægt på at beskrive udvidelsen af en eksisterende metode til estimering af parametre i kontinuert-diskrete stokastiske tilstandsmodeller, således at den egner sig bedre til modeller af fed-batch processer, samt på implementeringen af denne metode i et computerprogram kaldet **CTSM**, og det vises at dette program er væsentligt bedre, både med hensyn til estimaternes kvalitet og med hensyn til reproducerbarhed, end et andet program, der bygger på en lignende metode. Værktøjer implementeret i MATLAB, der kan bruges i forbindelse med andre trin i grey-box-modeldannelsescyklussen beskrives også, og baseret på samtlige de enkelte trin præsenteres en grey-box-modelleringsalgoritme, der kan bruges til systematisk iterativ forbedring af modeller, og dennes egenskaber og begrænsninger diskuteres herefter kort.

En særligt vigtig egenskab er, at grey-box-modelleringsalgoritmen bibringer en metodik, der kan bruges til at lokalisere mangler i modeller ved hjælp af information fra eksperimentelle data, hvorefter årsagen til disse mangler kan afdækkes på en måde, der giver et fingerpeg om, hvorledes modellen kan forbedres. Dette er en særdeles vigtig egenskab, som andre metoder til grey-box-modellering fra litteraturen ikke besidder, idet de i stedet er helt afhængige af brugerens evne til at foreslå modelforbedringer, hvorfor der kan argumenteres for, at den her foreslåede metode i denne henseende er mere systematisk, hvilket er et vigtigt resultat. På linie med andre metoder til grey-box-modellering er den her foreslåede metode dog begrænset af både mængden og kvaliteten af den a priori viden og de eksperimentelle data, der er til rådighed, så der gives også en diskussion af konsekvenserne heraf. Den foreslåede metodik illustreres via en række anvendelseseksempler, på basis af hvilke der argumenteres for, at grey-box-modelleringsalgoritmen faktisk kan konvergere for visse simple systemer, selvom der ikke findes noget stringent bevis for dette, samt for, at metodikken under alle omstændigheder gør modelopstillingsarbejdet lettere. Der præsenteres desuden en generaliseret udgave af grey-box-modelleringsalgoritmen, som ikke er begrænset til modellering af fed-batch processer med henblik på tilstandsestimering og optimal regulering, men som kan bruges mere generelt til modellering af en lang række systemer med henblik på forskellige formål.

Contents

1

Introduction

The purpose of this chapter is to motivate the work presented in this thesis, state the objective of the work and give a brief overview of the most important results. Since the primary focus of the work is on modelling of fed-batch processes for the purpose of state estimation and optimal control, Section 1.1 is devoted to establishing some basic principles for such processes. Within this section an introduction to modelling of fed-batch processes based on first engineering principles is given along with an outline of the state of the art of fed-batch process operation in industry. By means of a discussion of present shortcomings of the latter the motivation is given in Section 1.2 in terms of an expression of the need for an efficient approach to automatic fed-batch process operation and a list of requirements for such an approach. A review of various approaches reported in literature is also given along with a discussion of their merits with respect to meeting these requirements. This review serves to further motivate the work, the objective of which is stated in Section 1.3 in terms of a proposal for an alternative approach in the form of an overall framework for fed-batch process modelling, state estimation and optimal control based on grey-box models. Attention is then restricted to the modelling part of this framework, a description of which is also given, and based on this description, an overview of the most important results is given in Section 1.4. Finally, an outline of the contents of the remainder of the thesis is given in Section 1.5.

1.1 Preliminaries

Fed-batch processes are common in chemical industry, ranging from conventional semi-batch reactors in the specialty chemicals industry to fed-batch bioreactors in the biochemical and pharmaceutical industries, and they are characterized by taking place in a closed vessel and by running for a finite period of time or until a certain amount of product has been obtained. During the entire course of a fed-batch run new reactants are continuously fed to the vessel, but no products are taken out until the end, where the vessel is emptied and the contents led to downstream processing equipment. Fed-batch processing is often used when continuous processing is infeasible, the idea being to maintain some level of continuity in production by repeating the process.

1.1.1 Basic fed-batch process modelling

Within chemical engineering the derivation of mathematical process models is traditionally based on first engineering principles, which means that model development starts off from the general balance equation, i.e.:

$$\text{Accumulation} = \text{Input} + \text{Generation} - \text{Output} - \text{Consumption} \tag{1.1}$$

which applies to mass, energy and other conserved quantities for all types of processes and gives rise to a set of ordinary differential equations, i.e.:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \tag{1.2}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a vector of balanced quantities, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is a vector of input variables and $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, and where, in the general case, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$ is a *nonlinear* function. In addition to the above set of differential equations a number of implicit algebraic equations are usually needed, e.g. in order to describe the thermodynamics of the process.

Models of fed-batch processes are often linear in the input variable(s), which gives rise to a simpler set of ordinary differential equations, i.e.:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, t, \boldsymbol{\theta}) + \boldsymbol{g}(\boldsymbol{x}_t, t, \boldsymbol{\theta})\boldsymbol{u}_t \tag{1.3}$$

where $t \in [t_0, t_f] \subset \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a vector of balanced quantities, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is a vector of input variables and $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, and where $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$ and $\boldsymbol{g}(\cdot) \in \mathbb{R}^{n \times m}$ are *nonlinear* functions.

A model of this type is described in the following example, and, whenever possible, this simple model of a fed-batch fermentation process will be used to illustrate important concepts throughout the remainder of this thesis.

Example 1.1 (A model of a fed-batch fermentation process)
This example describes a simple model of a fed-batch fermentation process. Figure 1.1 shows a sketch of the process with a stream of medium, which consists of water and substrate, being fed to a stirred tank reactor containing fermentation broth, which consists of water, substrate and biomass. The model describes growth of biomass on a single substrate with Monod kinetics and substrate inhibition as follows:

$$\frac{dX}{dt} = \mu(S)X - \frac{FX}{V} \tag{1.4}$$

$$\frac{dS}{dt} = -\frac{\mu(S)X}{Y} + \frac{F\,(S_F - S)}{V} \tag{1.5}$$

$$\frac{dV}{dt} = F \tag{1.6}$$

for $t \in [t_0, t_f]$, where X $(\frac{g}{l})$ is the biomass concentration, S $(\frac{g}{l})$ is the substrate concentration, V (l) is the reactor volume, F $(\frac{l}{h})$ is the feed flow rate, $Y = 0.5$ is a yield coefficient and $S_F = 10\frac{g}{l}$ is the feed concentration of substrate. $t_0 = 0h$ and

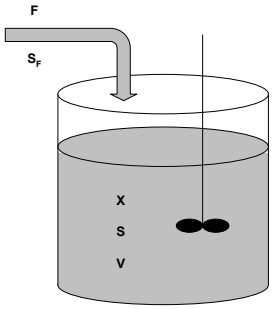


Figure 1.1. Simple sketch of a fed-batch bioreactor.

$t_f = 3.8h$ are initial and final times of a typical fed-batch run and $\mu(S)$ (h^{-1}) is the biomass growth rate, which can be represented by the following expression:

$$\mu(S) = \mu_{\max} \frac{S}{K_2 S^2 + S + K_1} \tag{1.7}$$

where $\mu_{\max} = 1h^{-1}$, $K_1 = 0.03\frac{g}{l}$ and $K_2 = 0.5\frac{l}{g}$ are kinetic parameters. The parameter values used correspond to the values used by Kuhlmann *et al.* (1998). ■

1.1.2 Fed-batch process operation

In industry fed-batch processes are repeated over and over again to maintain some level of continuity in production. To ensure uniform product quality and to ease the problem of overall scheduling in a plant with several pieces of processing equipment in series or parallel, it is desirable to have similar operating conditions every time a process is repeated. In other words one goal of fed-batch processing is *uniform operation*. Another goal, and a goal which is more difficult to achieve, is *optimal productivity*. The definition of productivity depends on the particular process. It is usually a function of the amount of product at the end of a run and the product quality and purity, but it may also be a function of the utilization of reactants or the formation of biproducts.

Determining operating conditions, which ensure uniform operation and optimal productivity, is very difficult, because it involves developing a sufficiently accurate mathematical model of the process, stating a reasonable optimisation problem and subsequently solving this problem. Three steps, which are all difficult in their own right, but which together and along with the limitations set by the fact that the real world is not ideal, pose a problem, which is almost impossible to solve. The best way to illustrate this is to give an example, showing how the solution to a particular productivity maximization problem can be used to determine the operating conditions for a fed-batch process in an ideal world, and subsequently explain why this approach fails in practice.

Example 1.2 (Optimal operation of the fermentation process)
The model described in Example 1.1 was used by Kuhlmann *et al.* (1998) in a simulation study of optimisation of fed-batch fermentation processes, where the objective was to optimize the production of biomass by manipulating the feed flow rate given a set of fixed initial conditions and constraints on the reactor volume and the feed flow rate. The present example illustrates how a relaxed version of this optimisation problem with manipulable initial conditions and without constraints can be solved analytically, as shown by Visser (1999). The problem can be stated as follows:

$$\max_{\substack{X_0,S_0,V_0,\\ F(t)\ ,\ t\in[t_0,t_f]}} V(t_f)X(t_f) \tag{1.8}$$

subject to:

$$\begin{array}{l} \dfrac{dX}{dt} = \mu(S)X - \dfrac{FX}{V} \\ \dfrac{dS}{dt} = -\dfrac{\mu(S)X}{Y} + \dfrac{F\,(S_F - S)}{V} \\ \dfrac{dV}{dt} = F \end{array} \quad , \quad \begin{array}{l} X(t_0) = X_0 \\ S(t_0) = S_0 \\ V(t_0) = V_0 \end{array} \quad , \, t \in [t_0, t_f] \tag{1.9}$$

where:

$$\mu(S) = \mu_{\max}\frac{S}{K_2S^2 + S + K_1} \tag{1.10}$$

In other words, the problem is to determine the initial conditions and the open loop feed flow rate trajectory that gives optimal productivity in terms of the amount of biomass at the end of a run. By applying an appropriate variable transformation and subsequently using Pontryagin's maximum principle, or by simply applying the intuitive argument that the productivity is maximized when the biomass growth rate is maximized, the following condition for optimal operation can be obtained:

$$0 = \frac{d\mu(S)}{dS} = \mu_{\max}\frac{K_1 - K_2S^2}{(K_2S^2 + S + K_1)^2} \Rightarrow S = \sqrt{\frac{K1}{K2}} = S^* \tag{1.11}$$

Assuming that the initial substrate concentration $S_0 = S^*$ and by choosing the feed flow rate in a way that makes $\frac{dS}{dt} = 0$, S can be kept at $S_0 = S^*$, i.e.:

$$0 = \frac{dS}{dt} = -\frac{\mu(S_0)X}{Y} + \frac{F\,(S_F - S_0)}{V} \Rightarrow F = \frac{\mu(S_0)XV}{Y\,(S_F - S_0)} \tag{1.12}$$

This expression is inserted into the other two equations of the original model, i.e.:

$$\begin{array}{l} \dfrac{dX}{dt} = \mu(S_0)X - \dfrac{\mu(S_0)XV}{Y\,(S_F - S_0)}\dfrac{X}{V} \\ \dfrac{dV}{dt} = \dfrac{\mu(S_0)XV}{Y\,(S_F - S_0)} \end{array} \quad , \quad \begin{array}{l} X(t_0) = X_0 \\ V(t_0) = V_0 \end{array} \quad , \, t \in [t_0, t_f] \tag{1.13}$$

and by setting $a = \mu(S_0)$ and $b = \frac{\mu(S_0)}{Y(S_F - S_0)}$, the equation for X can be solved:

$$\begin{array}{l} \dfrac{dX}{dt} = aX - bX^2 \\ X = \dfrac{ae^{at}c}{1 + be^{at}c} \end{array} \quad , \, t \in [t_0, t_f] \tag{1.14}$$

with $c = \frac{X_0}{a - bX_0}$, whereupon the equation for V can be solved as follows:

$$\begin{aligned} \frac{dV}{dt} &= bXV = b\frac{ae^{at}c}{1 + be^{at}c}V \\ V &= \frac{1 + be^{at}c}{1 + bc}V_0 \ , \ t \in [t_0, t_f] \end{aligned} \tag{1.15}$$

By substituting these solutions back into the equation for the feed flow rate, i.e.:

$$\begin{aligned} F = bXV &= b\frac{ae^{at}c}{1 + be^{at}c}\frac{1 + be^{at}c}{1 + bc}V_0 \\ &= be^{at}X_0V_0 \ , \ t \in [t_0, t_f] \end{aligned} \tag{1.16}$$

an analytical expression for the optimal feed flow rate trajectory can be obtained. ■

The example above shows how the solution to a particular productivity maximization problem can be used to determine the operating conditions for a process in an ideal world. However, the real world is not ideal, so in practice this approach fails. More specifically, the approach relies on the assumption that the model of the process is correct and that there are no disturbances. This is due to the fact that the feed flow rate trajectory is an open loop trajectory calculated off-line, meaning that no measures can be taken on-line to account for the effects of mismatch between the model and the actual process and for the effects of disturbances. In the real world fed-batch processes are always affected by disturbances, and no model can ever capture all the characteristics of a process. In other words an alternative approach, which is able to handle model uncertainty and disturbances, is needed. An essential part of such an approach is a feedback controller, which acts on measurements of process variables, but because measurements can only be obtained at discrete points in time, and because not all process variables can be measured, especially not on-line, the approach must be able to handle discretely, partially observed systems, and because the measurements that are available may be corrupted with measurement noise, the approach must be able to handle this as well.

A number of such approaches have been presented in literature, and some have even been successfully applied to laboratory scale processes. Unfortunately, industrial scale processes are more complicated and more difficult to control, e.g. due to operational limitations such as unknown initial conditions and state and input variable constraints, so very few of these approaches have been implemented in industry. Today most fed-batch processes in industry are therefore run by a human operator according to personal experience and rules of thumb, and as a result operation is not always uniform and optimal productivity is seldom obtained. More details about the state of the art of fed-batch process operation are given by Bonvin (1998) and Srinivasan *et al.* (2002a,b).

1.2 Motivation

From the discussion given in the previous section it is evident that there is a need for an efficient approach to operation of fed-batch processes, which will ensure *uniform operation* and *optimal productivity* in an automatic manner, i.e. without requiring the intervention of a human operator. Such an approach must be model-based and it must reflect the fact that fed-batch processes are inherently nonlinear. Furthermore, it must be able to handle model uncertainty and disturbances, even for discretely, partially observed systems with measurement noise. Finally, it must be able to handle operational limitations such as unknown initial conditions and state and input variable constraints.

The first step towards developing an approach that fulfills these objectives, is to decide how to model fed-batch processes. Should modelling be based on first engineering principles? Should it be data-driven? Or should it somehow be a combination of both of these approaches? This is discussed in the following.

1.2.1 First engineering principles modelling

Models based on first engineering principles are intuitively appealing in the way they are derived and in their ability to reflect the nonlinear nature of fed-batch processes. Most of the work that has been presented in literature on automatic operation of fed-batch processes is based on such models.

In early papers there was a tendency to assume ideal world conditions and concentrate on calculating optimal open loop input trajectories. An example by Visser (1999) of an analytical solution to a problem of this type has already been given. For more complicated systems, where no analytical solution exists, Cuthrell and Biegler (1989) have shown how to find a solution by applying orthogonal collocation, formulating a nonlinear program (NLP) and solving the NLP by applying successive quadratic programming (SQP). A detailed overview of both anatytical and numerical solution methods for such batch process optimisation problems is given by Srinivasan *et al.* (2002a).

More recently Ruppen *et al.* (1995) and Kuhlmann *et al.* (1998) have shown how to account for model uncertainty when determining optimal open loop input trajectories. An overview of these and similar methods for batch process optimisation under uncertainty is given by Srinivasan *et al.* (2002b).

These methods still fail to account for disturbances, however, and because predetermined uncertainty bounds are assumed, there is a risk of obtaining overly conservative input trajectories, but these problems can be solved by applying feedback control along the input trajectories as shown by Kuhlmann *et al.* (1998) and Visser (1999), and by using experimentally determined uncertainty bounds. Unfortunately, the latter is difficult due to the nonlinear nature of fed-batch processes, and both the former and the latter is complicated by the fact that such processes are examples of discretely, partially observed systems.

An alternative way to account for model uncertainty and disturbances that has also been reported in literature, is to apply robust control along open loop input trajectories determined without accounting for model uncertainty. It is very difficult to apply nonlinear robust control directly, so a two-loop controller with an inner-loop nonlinear linearizing controller and an outer-loop linear robust controller is often used to account for the nonlinear nature of fed-batch processes. Constructing a nonlinear linearizing controller involves complicated analytical manipulations based on Lie algebra to determine an expression for the nonlinear compensator, and evaluating the expression for the compensator usually requires current values of all state variables, so, although nonlinear observers can be designed to provide estimates of these for discretely, partially observed systems, this approach is unsuitable for industrial scale processes.

Adaptive control provides yet another way to account for model uncertainty and disturbances as shown by e.g. Dochain and Bastin (1988) and van Impe and Bastin (1995). The idea is to use the information that is obtained when determining open loop input trajectories to form model-independent heuristic control objectives that can easily be fulfilled by applying nonlinear linearizing control based on on-line state and parameter estimation. Unfortunately, relying on nonlinear linearizing control, this approach is hardly suitable for industrial scale processes either, but unlike the other approaches described here, it is able to handle discretely, partially observed systems with measurement noise.

The above approaches to automatic operation of fed-batch processes based on first engineering principles models all have obvious shortcomings. This indicates that, although intuitively appealing, such models are not necessarily adequate for modelling fed-batch processes for the purpose of automatic operation. Furthermore, first engineering principles models are time-consuming to develop, because few systematic methods are available for making inferences about the proper structure of such models, which can seldom be determined completely from prior physical knowledge, and because the parameters of such models can only be estimated from experimental data with parameter estimation methods that tend to give biased and unreproducible results, because random effects are absorbed into the parameter estimates. Data-driven models, for which systematic methods for structural identification and more appropriate parameter estimation methods are available, are therefore often used instead.

1.2.2 Data-driven modelling

Data-driven models are developed through identification experiments, usually in the form of input-output models. In principle, data-driven models include both nonparametric and parametric models and may be formulated in both continuous and discrete time, but discrete time parametric models are by far the most widely used, so for the purpose of the following discussion the term "data-driven models" means discrete time parametric input-output models.

Relying predominantly on data-based information and being sensitive to the quality of this information, data-driven models are not as appealing as first engineering principles models in terms of providing a consistent and physically meaningful system description, but they are easier to use for fed-batch process modelling, because their inherent input-output nature make them suitable for discretely, partially observed systems with measurement noise, and because their development through identification experiments allows statistical information about model uncertainty to be obtained directly and non-conservatively.

Unfortunately, nonlinear data-driven models, which most adequately reflect the nonlinear nature of fed-batch processes, are difficult and computationally burdensome to identify as discussed by Unbehauen (1996). Hence the amount of work that has been presented in literature on automatic operation of fed-batch processes with such models is not substantial. A larger amount of work has been presented with linear data-driven models, particularly for the purpose of monitoring but also for the purpose of automatic operation. A quite promising approach in this area has been proposed by Lee *et al.* (1999) and is based on exploiting the repetitive nature of fed-batch processes by combining iterative learning with a model predictive control (MPC) scheme for simultaneous trajectory tracking and quality control. Explaining in more detail, how this approach works, is quite involved, but the general idea is to make a model from run to run of the errors with respect to pre-determined reference trajectories and use this model along with information from previous runs and measurements from the current run to improve the performance of the current run. The most considerable advantage of this approach is its ability to handle processes with inherently nonlinear intra-run dynamics by instead modelling run-to-run dynamics in a linear fashion. Good results have been reported by Lee *et al.* (1999), showing the ability of this approach to improve the performance from run to run by decreasing the errors. The only problem is that pre-determined reference trajectories are needed. Such trajectories may be determined in two different ways. They may be specified by a human operator according to personal experience and rules of thumb, in which case the approach will guarantee uniform operation to a certain extent, but not optimal productivity. Alternatively, to achieve this, the necessary reference trajectories may be determined by solving an optimisation problem using a suitable intra-run model of the process, but finding a data-driven model for this purpose is difficult, because the model must be able to reflect the nonlinear nature of fed-batch processes.

Evaluating the usefulness of data-driven models, this is a serious drawback, as is the lack of appeal in terms of providing a consistent and physically meaningful system description as well as the sensitivity of data-driven models to the quality of the data-based information used for their development, because of the substantial influence it may have on the solution to an optimisation problem if the model being used is based on data obtained under non-optimal conditions, and this all indicates that data-driven models are not necessarily adequate for modelling fed-batch processes for the purpose of automatic operation either.

1.2.3 Hybrid modelling

With the above discussion in mind, it seems natural to combine first engineering principles modelling and data-driven modelling into a hybrid modelling scheme that takes advantage of the strenghts of both, and a number of such schemes, based on neural networks, have been developed within the last decade.

One of the first was proposed by Psichogios and Ungar (1992), who suggested to use neural networks to model the state-dependence of certain parameters of a first engineering principles model, e.g. the biomass growth rate in a model of a fed-batch bioreactor. The objective of their work was to develop a modelling scheme that was more flexible than classical parameter estimation schemes and more efficient than purely data-driven modelling, and judging from their simulation results, the proposed hybrid model performed very well in that respect. More specifically, without having to know the specific parameterization of the state-dependence of the biomass growth rate, and without having to train the neural network that was used instead for very long, the hybrid model was able to very accurately predict the evolution of the state variables.

Following the work by Psichogios and Ungar (1992) and work in the same area by Su *et al.* (1993), a number of different applications of hybrid modelling with neural networks have been reported, e.g. by Martinez and Wilson (1998), who successfully applied hybrid modelling to the optimisation of a batch unit.

A considerable advantage of hybrid modelling with neural networks is that it is relatively easy to use and therefore readily applicable to simple systems. For more complicated systems, however, extensive training data sets may be needed and determining a suitable model may be very time-consuming, particularly if the model elements modelled with neural networks depend on unmeasured state variables, or if the measurements are corrupted with noise. This in turn stresses the need to find other modelling approaches that are more adequate for modelling fed-batch processes for the purpose of automatic operation.

1.2.4 Grey-box modelling

One such approach, and another approach that provides an appealing trade-off between first engineering principles modelling and data-driven modelling, is grey-box modelling (Madsen and Melgaard, 1991; Melgaard and Madsen, 1993; Bohlin and Graebe, 1995; Bohlin, 2001), which aims at developing stochastic state space models consisting of a set of stochastic differential equations (SDE's) describing the dynamics of the system in continuous time and a set of discrete time measurement equations. The key idea of grey-box modelling is to find the simplest model for a given purpose, which is consistent with prior physical knowledge and not falsified by available experimental data. In the specific approach by Bohlin and Graebe (1995) and Bohlin (2001), this is done by formulating a sequence of hypothetical model structures of increasing complexi-

ty and systematically expanding the model by falsifying incorrect hypotheses through statistical tests based on the experimental data. A major advantage of this approach is that by proper selection of these tests, models can be developed with different properties, e.g. in terms of prediction capabilities, which means that models can be designed specifically to serve a given purpose, including automatic operation of fed-batch processes. A drawback is that it is an iterative and inherently interactive approach, because it relies on the model maker to formulate the hypothetical model structures to be tested, which poses the problem that the model maker may run out of ideas for improvement before a sufficiently accurate model is obtained. However, the advantages of grey-box modelling seem to outweigh the drawbacks, for which reason this is the approach that has been further pursued in the work presented in this thesis.

Grey-box models are designed to accomodate random effects and allow for a decomposition of the noise affecting the system into a process noise term and a measurement noise term. As a consequence of this *prediction error decomposition* (PED), unknown parameters of such models can be estimated from experimental data in a *prediction error* (PE) setting (Young, 1981) as is the case for data-driven models, whereas for first engineering principles models it can only be done in an *output error* (OE) setting (Young, 1981), which tends to give biased and less reproducible results, because random effects are absorbed into the parameter estimates. Furthermore, PE estimation allows for subsequent application of a number of powerful statistical tools to provide indications for possible model improvements. In fact, one of the key results of the work presented in this thesis is that, by proper application of such tools, grey-box modelling can be made more systematic and less reliant on the model maker than in the approach by Bohlin and Graebe (1995) and Bohlin (2001).

1.3 Objective

As indicated in the previous section there is a need to find new modelling approaches, which are suited for automatic operation of fed-batch processes with the aim of achieving uniform operation and optimal productivity. The work presented in this thesis focuses on this issue, and the objective of the work has been to develop a systematic grey-box modelling framework for fed-batch process modelling for the purpose of automatic operation. However, because the models developed within this framework must be applicable in the context of an appropriate overall framework for automatic operation, which is able to fulfill the goals of uniform operation and optimal productivity, the main features of such a framework have also been established. In the following an overall framework for fed-batch process modelling, state estimation and optimal control is therefore briefly outlined before attention is restricted to the systematic grey-box modelling framework being proposed in this thesis.

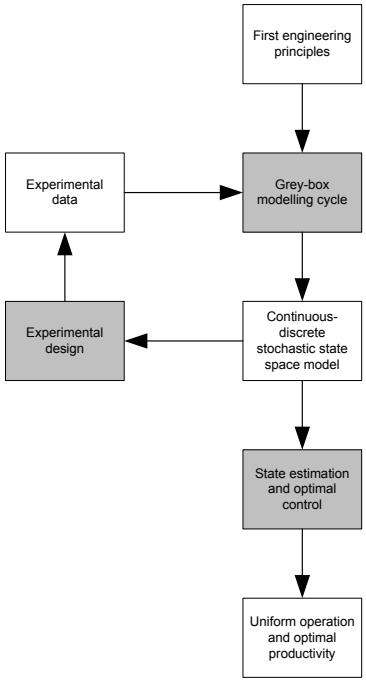


Figure 1.2. An overall framework for fed-batch process modelling, state estimation and optimal control incorporating the proposed grey-box modelling framework.

1.3.1 Description of the overall framework

The overall framework is best described by considering Figure 1.2, which shows the individual elements and how they are interrelated. Elements shown in grey constitute *tasks* and elements shown in white constitute various items that serve as input to or output from the individual tasks of the framework. The first and most comprehensive of these tasks is the *grey-box modelling cycle*, which constitutes the proposed grey-box modelling framework. A more detailed outline of this framework is given later, but it serves to combine first engineering principles modelling with data-driven modelling and therefore has two inputs in the form of *first engineering principles* and *experimental data*, and the output from the task is a *continuous-discrete stochastic state space model*, which serves as input to the remaining tasks of the overall framework. A continuous-discrete stochastic state space model consists of a continuous time system equation given by a set of SDE's and a discrete time measurement equation given by a set of algebraic equations. The system equation can be formulated as follows:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt + \boldsymbol{\sigma}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{1.17}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a vector of state variables, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is a vector of input variables, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$ and $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ are nonlinear functions and $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process. The measurement equation can be formulated as follows:

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{1.18}$$

where $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is a vector of output variables, $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ a nonlinear function and $\{\boldsymbol{e}_k\}$ an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

Assumption no. 1. Since, as previously mentioned, models of fed-batch processes are often linear in the input variable(s), it is assumed throughout the remainder of this thesis that a simplified version of the general formulation can be used. The simplified system equation can be formulated as follows:

$$d\boldsymbol{x}_t = (\boldsymbol{f}(\boldsymbol{x}_t, t, \boldsymbol{\theta}) + \boldsymbol{g}(\boldsymbol{x}_t, t, \boldsymbol{\theta})\boldsymbol{u}_t)dt + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{1.19}$$

where $t \in [t_0, t_f] \subset \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a state vector, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is an input vector, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{g}(\cdot) \in \mathbb{R}^{n \times m}$ and $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ are nonlinear functions and $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process. The measurement equation remains the same, i.e.:

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{1.20}$$

where $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is a vector of output variables, $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ a nonlinear function and $\{\boldsymbol{e}_k\}$ an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

Assumption no. 2. For the purpose of simplicity it is also assumed that additional implicit algebraic equations are not needed. As discussed in Chapter 5, relaxation of this assumption is a very important possible topic for future work.

Having established what is meant by the continuous-discrete stochastic state space model generated as an output from the grey-box modelling cycle, the remaining tasks of the overall framework can be explained. The task labeled *experimental design* deals with design of identification experiments, i.e. with how to perform experiments on a given process in a way that provides optimal information under given circumstances. The model serves as input to this task, because experimental design is highly dependent on the model to be identified, and the output from the task is experimental data, implying that performing experiments is also a part of this task. The experimental data serve as input to the grey-box modelling cycle, hereby closing the loop shown in Figure 1.2, the idea of which is to indicate the possibility of repeatedly using the grey-box modelling cycle and the experimental design task to iteratively improve the quality of the model. This issue is outside the scope of the work presented in this thesis, but it is a very important possible topic for future work, as discussed in Chapter 5. Once the quality of the continuous-discrete stochastic state space model is satisfactory, the *state estimation and optimal control* task can be executed, and, by using the model as input, the idea of this task is to design

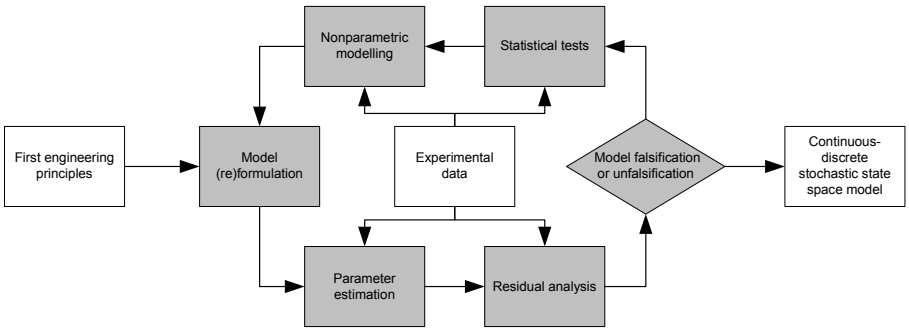


Figure 1.3. The grey-box modelling cycle of the overall framework.

optimal multivariable control, e.g. MPC, with simultaneous state estimation to achieve the goals of *uniform operation and optimal productivity*. As discussed in more detail later, continuous-discrete stochastic state space models have several attractive features in this regard, but the issue of developing specific methods for optimal control with simultaneous state estimation based on such models is outside the scope of the work presented in this thesis. Instead, this is another very important possible topic for future work, as discussed in Chapter 5.

1.3.2 Description of the grey-box modelling cycle

Returning to the grey-box modelling cycle, which is the main topic of the remainder of this thesis, it is best described by considering Figure 1.3, which shows its individual elements and how they are interrelated. Again, elements shown in grey constitute *tasks* and elements shown in white constitute various input and output items that have already been described. The idea of the first task, i.e. the *model (re)formulation* task, is to use first engineering principles and all other relevant prior physical knowledge to construct an initial continuous-discrete stochastic state space model, or at least to establish the basic structure of such a model. In the *parameter estimation* task the idea then is to estimate the parameters of this model from experimental data using an appropriate parameter estimation method. On the basis of these estimates and more experimental data, the idea of the *residual analysis* task then is to perform cross-validation residual analysis to obtain information about the quality of the resulting model. Based on this information, the idea of the *model falsification or unfalsification* task then is to determine whether or not the model is sufficiently accurate for the purpose of state estimation and optimal control. If this is the case, the model is said to be *unfalsified* with respect to the available information and the model development procedure implied by the grey-box modelling cycle can be terminated, wherupon the model can be used as input to the state estimation and optimal control task. If, on the other hand,

the model is *falsified*, the model development procedure must be repeated, and the idea of the *statistical tests* task then is to use statistical tests to pinpoint deficiencies within the model, if this possible. If this is the case, the idea of the *nonparametric modelling* task then is to determine how to repair these deficiencies by applying nonparametric methods and subsequently using the resulting information to alter the model in accordance with available physical knowledge. Hereby returning to the model re(formulation) task, the loop shown in Figure 1.3 is closed, the idea of which is to indicate the possibility of iteratively improving the quality of the model given a fixed amount of experimental data, until the model is unfalsified, or at least until no more information can be extracted from the experimental data. In the latter case the model remains falsified until more information becomes available, e.g. in the form of new experimental data obtained from specifically designed experiments, as discussed above. The individual tasks of the grey-box modelling cycle are described in much more detail in Chapter 2, where an algorithm for systematic iterative model improvement based on the grey-box modelling cycle is also presented.

1.3.3 Justification for the overall framework

The following discussion serves to justify the overall framework for fed-batch process modelling, state estimation and optimal control described in this section as being a powerful alternative to the various other approaches to automatic operation of fed-batch processes described in the previous section.

An advantage of the overall framework described here is that it combines first engineering principles modelling with data-driven modelling in a way that retains the intuitive appeal of first engineering principles models in terms of their derivation and physical interpretability, and at the same time allows iterative model improvement based on the principles of data-driven modelling, both with a fixed amount of experimental data and in an iterative scheme that includes experimental design and facilitates run-to-run updating of the model.

Moreover, the continuous-discrete stochastic state space model has a number of attractive features of its own with respect to the requirements stated in the previous section: It is able to reflect the nonlinear nature of fed-batch processes, the SDE's in the continuous time system equation (1.19) enables it to handle uncertainty and disturbances through the diffusion term (the second term), and the discrete time measurement equation (1.20) enables it to handle discretely, partially observed systems with measurement noise in a sensible manner.

The overall framework described here also has the advantage of facilitating estimation of the parameters of the diffusion term of the system equation and the noise term of the measurement equation, which in turn allows model uncertainty, disturbances and measurement noise to be handled in a non-conservative way, which is very important when subsequently using the model for state estimation and optimal control. Continuous-discrete stochastic state space models

are very easy to use for state estimation, and having estimated the parameters of the diffusion term and the measurement noise term it is believed that better estimates can be obtained than otherwise. Designing optimal multivariable control based on such models is also believed to be relatively straightforward, e.g. by means of MPC, which will allow operational limitations such as state and input variable constraints to be taken into account as well. As mentioned, a thorough investigation of these issues is outside the scope of the work presented in this thesis, and the discussion given here merely serves to justify the efforts put into developing the proposed grey-box modelling framework.

1.4 Overview of results

The work presented in this thesis has been application-oriented in the sense that, instead of rigorous theoretical developments, the primary focus has been on development of the proposed grey-box modelling framework and in particular on the development of a number of simple methods and tools for facilitating the individual tasks within the grey-box modelling cycle shown in Figure 1.3.

1.4.1 Methods

In terms of methods, the primary result is the grey-box modelling cycle as a whole, because it provides a methodology for development of models of fed-batch processes for the purpose of state estimation and optimal control.

A key feature in this regard is that the methodology facilitates systematic pinpointing of model deficiencies based on information extracted from experimental data and allows the structural origin of these deficiencies to be uncovered as well to provide guidelines for model improvement. This is a very powerful feature not shared by other approaches to grey-box modelling reported in literature, which rely solely on the model maker to determine how to improve the model. In other words, the proposed methodology is more systematic and less reliant on the model maker, which is a key result, as is the fact that this methodology is not limited to modelling of fed-batch processes for the purpose of state estimation and optimal control but can be generalized into a version that can be applied to model a variety of systems for different purposes.

Another significant but much more technical result with respect to methods is the extension of an existing parameter estimation method for continuous-discrete stochastic state space models by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) to make it more readily applicable to models of fed-batch processes. In particular the inability of the original method to handle models with singular Jacobians has been remedied and the method has been extended to allow estimation with multiple independent sets of experimental data and to handle missing observations in a much more appropriate way.

1.4.2 Tools

In terms of tools, the aforementioned parameter estimation method has been implemented in a computer program called **CTSM**, which is based on a similar program by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) called CTLSM. For ease of use this program has been equipped with a graphical user interface, and for the purpose of computational efficiency the binary code of the program has been optimized and prepared for shared memory parallel computing. With respect to this program an important result is that it has proven superior, both in terms of quality of estimates and in terms of reproducibility, to another program implementing a similar estimation method by Bohlin and Graebe (1995) and Bohlin (2001). In particular, more accurate and more consistent estimates of the parameters of the diffusion term can be obtained, which is important in the context of the grey-box modelling cycle.

A number of additional tools that facilitate other tasks within the grey-box modelling cycle, e.g. residual analysis, statistical tests and nonparametric modelling, have also been developed. These have been implemented in MATLAB.

1.5 Outline

The remainder of the thesis falls in three parts: A number of ordinary chapters, where rigorous mathematical details are omitted; a number of appendices, where these details are given; and two appendices containing selected papers.

In Chapter 2 the individual elements of the grey-box modelling cycle are described in detail and illustrated with examples, and a grey-box modelling algorithm that facilitates systematic iterative model improvement based on these elements is presented; Chapter 3 contains a number of examples of application of this algorithm; the conclusions are presented in Chapter 4; and a discussion of a number of possible topics for future work is given in Chapter 5.

In Appendix A a complete mathematical outline of the algorithms of the computer program **CTSM** is given; Appendix B contains an outline of the mathematical details of some statistical tests and residual analysis tools; and similar key details of some nonparametric methods are outlined in Appendix C.

The paper included in Appendix D contains the comparison mentioned above between **CTSM** and a program implementing a similar estimation method by Bohlin and Graebe (1995) and Bohlin (2001); and in the paper included in Appendix E a condensed outline of the grey-box modelling cycle and the corresponding algorithm is given with no particular emphasis on fed-batch process modelling. There is significant overlap between these papers and other parts of the thesis, but the papers also contain important additional results.

2

Methodology

In this chapter an outline of the proposed grey-box modelling framework is given by means of a description of the individual elements of the grey-box modelling cycle shown in Figure 1.3 and the concepts, theories and methods behind them. An algorithm for systematic iterative model improvement based on this modelling cycle is also presented. Whenever possible, rigorous mathematical details are omitted and instead given in the appropriate appendices.

2.1 Model (re)formulation

As discussed in Chapter 1, a key idea of grey-box modelling is to combine conventional model development based on first engineering principles and prior physical insights with statistical methods for structural identification, parameter estimation and model quality evaluation. This combination is facilitated by the use of continuous-discrete stochastic state space models, and the first element of the grey-box modelling cycle therefore deals with formulation of the initial structure of such a model. More specifically, this is a two-step procedure, where an ODE model is first derived from first engineering principles and then translated into a continuous-discrete stochastic state space model.

Deriving an ODE model of a fed-batch process from first engineering principles is a standard discipline, and, as shown in Section 1.1 (with the assumptions made in Section 1.3), this gives rise to a model of the following type:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, t, \boldsymbol{\theta}) + \boldsymbol{g}(\boldsymbol{x}_t, t, \boldsymbol{\theta})\boldsymbol{u}_t \tag{2.1}$$

where $t \in [t_0, t_f] \subset \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a vector of state variables, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is a vector of input variables, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, and $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$ and $\boldsymbol{g}(\cdot) \in \mathbb{R}^{n \times m}$ are nonlinear functions.

Translating the ODE model into a continuous-discrete stochastic state space model is also relatively straightforward, because it can be done by replacing the ODE's with appropriate SDE's and adding a set of algebraic equations

describing how measurements are obtained at discrete time instants. As shown in Section 1.3, this gives rise to a model of the following type:

$$d\boldsymbol{x}_t = (\boldsymbol{f}(\boldsymbol{x}_t, t, \boldsymbol{\theta}) + \boldsymbol{g}(\boldsymbol{x}_t, t, \boldsymbol{\theta})\boldsymbol{u}_t)dt + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{2.2}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{2.3}$$

where $t \in [t_0, t_f] \subset \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a state vector, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is an input vector, $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is an output vector, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{g}(\cdot) \in \mathbb{R}^{n\times m}$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n\times n}$ and $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ are nonlinear functions, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

In principle, any parameterization of $\boldsymbol{\sigma}(\cdot)$ can be used, but as shown in Section 2.5 using a diagonal parameterization has the advantage of facilitating pinpointing of model deficiencies, which is a key feature of the proposed grey-box modelling framework. A diagonal parameterization is therefore also used in the following example, which illustrates the above procedure for translating an ODE model into a continuous-discrete stochastic state space model.

Example 2.1 (Re-formulating the model of the fermentation process)
This example illustrates how the fermentation process model described in Example 1.1 can be translated into a continuous-discrete stochastic state space model. First the ODE's of the model are replaced with SDE's to give the system equation:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \ , \ t \in [t_0, t_f] \tag{2.4}$$

where σ_{11}, σ_{22} and σ_{33} are noise parameters. All other parameters, state and input variables are the same as in Example 1.1, and the biomass growth rate is given by:

$$\mu(S) = \mu_{\max}\frac{S}{K_2S^2 + S + K_1} \tag{2.5}$$

Then, assuming that all state variables can be measured directly at discrete time instants, a set of algebraic equations is added to give the measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{2.6}$$

where y_1, y_2 and y_3 are output variables. S_{11}, S_{22} and S_{33} are noise parameters. ■

As a matter of fact, the notation used for the SDE's in (2.2) is ambiguous unless a specific integral interpretation is given, so to resolve this issue and to establish some basic theoretical concepts, the remainder of this section is devoted to giving an introduction to SDE's and how they can be applied.

2.1.1 An introduction to SDE's

The use of SDE's is complicated by the advanced probability theory involved and by the fact that ordinary rules of calculus cannot always be applied. The following is therefore by no means a complete account of the theory behind SDE's but merely establishes the basic concepts. A much more detailed and mathematically rigorous introduction is given by Øksendal (1998).

The basis for the development of an SDE is the desire to include a stochastic part in an ODE to account for random effects. Starting from a simple ODE:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, t) \ , \ t \geq 0 \tag{2.7}$$

where $\boldsymbol{x}_t \in \mathbb{R}^n$ is a vector of state variables and $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$ is a nonlinear function, a first attempt might be to simply add noise to the equation to yield:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, t) + \boldsymbol{\sigma}(\boldsymbol{x}_t, t)\boldsymbol{w}_t \ , \ t \geq 0 \tag{2.8}$$

where $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ is a nonlinear function and $\{\boldsymbol{w}_t\}$ is a suitable stochastic process. Using this approach $\frac{d\boldsymbol{x}_t}{dt}$ becomes a random variable, and, if (2.8) is to retain the state property of (2.7), where the rate of change of the state variables is uniquely determined by their current values, the probability density of $\frac{d\boldsymbol{x}_t}{dt}$ must be uniquely determined by these values (Åström, 1970). This means that the stochastic process $\{\boldsymbol{w}_t\}$ must have the following properties:

- $\boldsymbol{w}_t$ is independent of $\boldsymbol{w}_s$ for $t \neq s$.
- $\{\boldsymbol{w}_t\}$ is stationary, i.e. $E\{\boldsymbol{w}_t \boldsymbol{w}_t^T\} < \infty$ for $t \geq 0$.
- $\boldsymbol{w}_t$ has zero mean for $t \geq 0$, i.e. $E\{\boldsymbol{w}_t\} = \boldsymbol{0}$ for $t \geq 0$.

but no "reasonable" such process exists[1], because it cannot have continuous paths (Øksendal, 1998). Thus (2.8) makes no sense (Åström (1970) argues that $\frac{d\boldsymbol{x}_t}{dt}$ cannot be expected to exist for a stochastic state space model) and an alternative way of including noise is needed. As it turns out, a more successful alternative is to subdivide the time interval $[0, t]$ as follows:

$$0 = t_0 < t_1 < \cdots < t_j < \cdots < t_{T-1} < t_T = t \tag{2.9}$$

and consider a discretized version of (2.8):

$$\boldsymbol{x}_{j+1} - \boldsymbol{x}_j = \boldsymbol{f}(\boldsymbol{x}_j, t_j)\Delta t_j + \boldsymbol{\sigma}(\boldsymbol{x}_j, t_j)\boldsymbol{w}_j \Delta t_j \ , \ j = 0, \ldots, T-1 \tag{2.10}$$

where $\boldsymbol{x}_j = \boldsymbol{x}_{t_j}$, $\boldsymbol{w}_j = \boldsymbol{w}_{t_j}$ and $\Delta t_j = t_{j+1} - t_j$, and then try to replace $\boldsymbol{w}_j \Delta t_j$ with $\Delta \boldsymbol{\omega}_j = \boldsymbol{\omega}_{j+1} - \boldsymbol{\omega}_j$, where $\{\boldsymbol{\omega}_t\}$ is a suitable stochastic process. The only

[1] As a matter of fact, it is possible to represent $\{\boldsymbol{w}_t\}$ by means of a so-called *generalized white noise process*, but this is not an ordinary stochastic process (Øksendal, 1998).

such process with continuous paths is the standard Wiener process (Øksendal, 1998), which is a mathematical description of the physical process of *Brownian motion*[2]. This process has the following important properties:

- $\boldsymbol{\omega}_0 = \mathbf{0}$ w.p. 1.
- $\{\boldsymbol{\omega}_t\}$ has continuous paths.
- $\boldsymbol{\omega}_t$ is Gaussian for $t \geq 0$.
- $\{\boldsymbol{\omega}_t\}$ has stationary independent increments.
- $\boldsymbol{\omega}_t$ has zero mean for $t \geq 0$, i.e. $E\{\boldsymbol{\omega}_t\} = \mathbf{0}$ for $t \geq 0$.

An important consequence of these properties is that an increment $\boldsymbol{\omega}_t - \boldsymbol{\omega}_s$, $0 \leq s < t$, of a standard Wiener process has the following properties:

- $\boldsymbol{\omega}_t - \boldsymbol{\omega}_s$ is Gaussian.
- $E\{\boldsymbol{\omega}_t - \boldsymbol{\omega}_s\} = \mathbf{0}$.
- $V\{\boldsymbol{\omega}_t - \boldsymbol{\omega}_s\} = (t-s)\boldsymbol{I}$.

Returning to (2.10) and replacing $\boldsymbol{w}_j \Delta t_j$ with $\Delta\boldsymbol{\omega}_j = \boldsymbol{\omega}_{j+1} - \boldsymbol{\omega}_j$, where $\{\boldsymbol{\omega}_t\}$ is a standard Wiener process, the following result can be obtained:

$$\boldsymbol{x}_T = \boldsymbol{x}_0 + \sum_{j=0}^{T-1} \boldsymbol{f}(\boldsymbol{x}_j, t_j)\Delta t_j + \sum_{j=0}^{T-1} \boldsymbol{\sigma}(\boldsymbol{x}_j, t_j)\Delta\boldsymbol{\omega}_j \tag{2.11}$$

and, by letting $\Delta t_j \to 0$, the following integral notation can be used:

$$\boldsymbol{x}_t = \boldsymbol{x}_0 + \int_0^t \boldsymbol{f}(\boldsymbol{x}_s, s)ds + \int_0^t \boldsymbol{\sigma}(\boldsymbol{x}_s, s)d\boldsymbol{\omega}_s \tag{2.12}$$

because it can be proven that the limit of the right-hand side of (2.11) exists if an appropriate interpretation of the second integral is given (Øksendal, 1998).

There are, however, different such interpretations, which in the general case yield different results. More specifically, to give an interpretation of the integral:

$$\int_0^t \boldsymbol{\sigma}(\boldsymbol{x}_s, s)d\boldsymbol{\omega}_s \tag{2.13}$$

it is defined as the limit, in a particular sense (Øksendal, 1998), of:

$$\sum_{j=0}^{T-1} \boldsymbol{\sigma}(\boldsymbol{x}_j^*, t_j^*)\Delta\boldsymbol{\omega}_j = \sum_{j=0}^{T-1} \boldsymbol{\sigma}(\boldsymbol{x}_j^*, t_j^*)(\boldsymbol{\omega}_{j+1} - \boldsymbol{\omega}_j) \text{ , for } T \to \infty \tag{2.14}$$

[2] *Brownian motion* refers to the characteristic, very irregular, motion of small particles dispersed in a fluid, and was first discovered in 1827 by scottish botanist Robert Brown.

where, depending on the particular choice of t_j^* in the interval $[t_j, t_{j+1}]$, different interpretations can be obtained, which yield different results:

- Choosing the left end point of the interval, i.e. $t_j^* = t_j$, gives rise to the so-called *Itô stochastic integral*.
- Choosing the middle of the interval, i.e. $t_j^* = \frac{t_j + t_{j+1}}{2}$, gives rise to the so-called *Stratonovich stochastic integral*.

As argued by Jazwinski (1970), neither of the two stochastic integrals is "right" nor "wrong", because they are simply different definitions. In fact there is an equivalent Itô integral for every Stratonovich integral[3] and all results for Stratonovich integrals have been proven with the theory for Itô integrals.

Unlike the Itô integral, which requires specialized stochastic calculus as shown below, the Stratonovich integral has the advantage of adhering to the ordinary rules of calculus in terms of facilitating integration by parts, variable substitution and application of the chain rule. However, the Itô integral is defined for a broader class of functions and has some nice mathematical properties not possessed by the Stratonovich integral, which make it more appropriate for filtering purposes (Jazwinski, 1970) and also for parameter estimation.

For this reason the Itô interpretation is used throughout this thesis. More specifically, whenever the following shorthand notation is used:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, t)dt + \boldsymbol{\sigma}(\boldsymbol{x}_t, t)d\boldsymbol{\omega}_t \ , \ t \geq 0 \tag{2.15}$$

it means that $\boldsymbol{x}_t$ is a solution to the corresponding integral equation:

$$\boldsymbol{x}_t = \boldsymbol{x}_0 + \int_0^t \boldsymbol{f}(\boldsymbol{x}_s, s)ds + \int_0^t \boldsymbol{\sigma}(\boldsymbol{x}_s, s)d\boldsymbol{\omega}_s \tag{2.16}$$

where the second integral is an Itô integral. Furthermore, since the two terms in (2.15) are commonly referred to as the *drift* term and the *diffusion* term respectively, this terminology is adapted throughout the thesis as well.

2.1.2 Itô stochastic calculus

The Itô integral requires specialized stochastic calculus. In the following a few important properties of Itô integrals and some rules from Itô stochastic calculus are therefore given. A more thorough outline is given by Øksendal (1998).

Assuming that $\boldsymbol{\sigma}(s)$, $\boldsymbol{\sigma}_1(s)$ and $\boldsymbol{\sigma}_2(s)$ are functions satisfying appropriate conditions (Øksendal, 1998), the following rules apply for Itô stochastic integrals:

$$\int_a^b \boldsymbol{\sigma}(s)d\boldsymbol{\omega}_s = \int_a^c \boldsymbol{\sigma}(s)d\boldsymbol{\omega}_s + \int_c^b \boldsymbol{\sigma}(s)d\boldsymbol{\omega}_s \tag{2.17}$$

[3]The two integrals actually coincide if $\boldsymbol{\sigma}(\cdot)$ does not depend on $\boldsymbol{x}_t$ (Øksendal, 1998).

$$\int_a^b \alpha\boldsymbol{\sigma}_1(s) + \beta\boldsymbol{\sigma}_2(s)d\boldsymbol{\omega}_s = \alpha\int_a^b \boldsymbol{\sigma}_1(s)d\boldsymbol{\omega}_s + \beta\int_a^b \boldsymbol{\sigma}_2(s)d\boldsymbol{\omega}_s \tag{2.18}$$

where $0 \leq a < c < b$, $\alpha \in \mathbb{R}$ and $\beta \in \mathbb{R}$. Expectations of Itô integrals are very important for many purposes and the following rules apply in this regard:

$$E\left\{\int_a^b \boldsymbol{\sigma}(s)d\boldsymbol{\omega}_s\right\} = \mathbf{0} \tag{2.19}$$

$$E\left\{\left(\int_a^b \boldsymbol{\sigma}(s)d\boldsymbol{\omega}_s\right)\left(\int_a^b \boldsymbol{\sigma}(s)d\boldsymbol{\omega}_s\right)^T\right\} = \int_a^b E\{\boldsymbol{\sigma}(s)\boldsymbol{\sigma}(s)^T\}ds \tag{2.20}$$

$$E\left\{\left(\int_a^b \boldsymbol{\sigma}_1(s)d\boldsymbol{\omega}_s\right)\left(\int_a^b \boldsymbol{\sigma}_2(s)d\boldsymbol{\omega}_s\right)^T\right\} = \int_a^b E\{\boldsymbol{\sigma}_1(s)\boldsymbol{\sigma}_2(s)^T\}ds \tag{2.21}$$

where the second rule is called the *Itô isometry* and is particularly important for filtering purposes (Jazwinski, 1970). Another very important rule is the so-called *Itô formula*, which is an Itô integral version of the chain rule and applies to a scalar function $\varphi(\boldsymbol{x}_t, t)$, where $\boldsymbol{x}_t$ is a solution to (2.15), as follows:

$$d\varphi = \left(\frac{\partial\varphi}{\partial t} + \frac{\partial\varphi}{\partial\boldsymbol{x}_t^T}\boldsymbol{f} + \frac{1}{2}\mathrm{tr}(\boldsymbol{\sigma}\boldsymbol{\sigma}^T\frac{\partial^2\varphi}{\partial\boldsymbol{x}_t\partial\boldsymbol{x}_t^T})\right)dt + \frac{\partial\varphi}{\partial\boldsymbol{x}_t^T}\boldsymbol{\sigma}d\boldsymbol{\omega}_t \tag{2.22}$$

where the shorthand notation $\varphi = \varphi(\boldsymbol{x}_t, t)$, $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{x}_t, t)$ and $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{x}_t, t)$ has been applied. Based on the Itô formula, stochastic versions of the rule of integration by parts and other standard rules can be derived (Øksendal, 1998).

2.1.3 Numerical solution of SDE's

Analytical solutions to SDE's are seldom available and numerical solution methods are therefore needed in most cases. A detailed account of a variety of such methods is given by Kloeden and Platen (1992), and the following is merely an introduction to some very simple discrete time approximation methods for simulation of SDE's, one of which is applied to generate the simulated data sets used in the examples presented throughout this thesis.

A number of discrete time approximation methods are available, which are all based on the *stochastic Taylor expansion*. The stochastic Taylor expansion resembles the conventional Taylor expansion, but is based on repeated application of the Itô formula. Different discrete time approximations with different orders of convergence can be obtained by using different numbers of terms in the stochastic Taylor expansion (Kloeden and Platen, 1992). The most simple of these methods is the *Euler scheme*, which can be used to simulate the solution to (2.15) by providing discrete time values $\boldsymbol{x}_j$, $j = 0, \ldots, T$, as follows:

$$\boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \boldsymbol{f}(\boldsymbol{x}_j, t_j)\Delta t_j + \boldsymbol{\sigma}(\boldsymbol{x}_j, t_j)\Delta\boldsymbol{\omega}_j \tag{2.23}$$

where $\Delta t_j = t_{j+1} - t_j$ is the discretization time interval and $\Delta\boldsymbol{\omega}_j = \boldsymbol{\omega}_{t_{j+1}} - \boldsymbol{\omega}_{t_j}$ is an $N(\mathbf{0}, \Delta t_j \boldsymbol{I})$ increment of the standard Wiener process. The error of this approximation is proportional to the square root of the size of the discretization time interval, and the method is therefore said to be strongly convergent of the order 0.5. An almost as simple scheme that is strongly convergent of the order 1.0 is the *Milstein scheme*, which, however, coincides with the Euler scheme if the diffusion term is independent of the state variables. Due to the assumptions made in Section 1.3 this is the case for the models considered in this thesis, and the Euler scheme is therefore applied to generate simulated data sets for the examples presented here. This is illustrated in the following example.

Example 2.2 (Generating data with the fermentation process model)
This example illustrates how the Euler scheme can be applied to simulate the solution to the system equation of the re-formulated model of the fermentation process shown in Example 2.1 to facilitate subsequent data generation with the complete continuous-discrete stochastic state space model (by sampling from the simulated solution with the measurement equation). Starting from appropriate initial states (X_0, S_0, V_0), the solution to the system equation of the model can be simulated as follows:

$$\begin{pmatrix} X \\ S \\ V \end{pmatrix}_{j+1} = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_{j} + \begin{pmatrix} \mu(S_j)X_j - \frac{F_j X_j}{V_j} \\ -\frac{\mu(S_j)X_j}{Y} + \frac{F_j(S_F - S_j)}{V_j} \\ F_j \end{pmatrix} \Delta t_j + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} \Delta\boldsymbol{\omega}_j \tag{2.24}$$

$$\mu(S_j) = \mu_{\max} \frac{S_j}{K_2 S_j^2 + S_j + K_1} \tag{2.25}$$

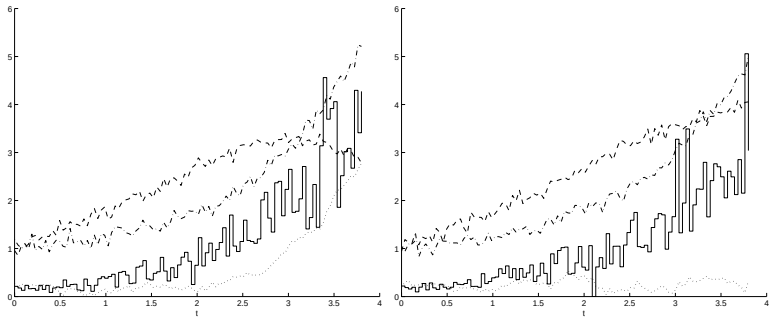
for $j = 0, \ldots, T$, by using $\Delta t_j = \frac{t_f}{T}$, $\Delta\boldsymbol{\omega}_j \in N(\mathbf{0}, \Delta t_j \boldsymbol{I})$ and appropriate values F_j for the feed flow rate. Subsequently, a set of observations can be generated by sampling from the simulated solution with the measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\mathbf{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{2.26}$$

for $k = 0, \ldots, N$. Using the initial states $(X_0, S_0, V_0) = (1, S^*, 1)$ and perturbed versions of the optimal feed flow rate trajectory determined in Example 1.2, a number of such data sets (shown in Figures 2.1-2.3) are generated for subsequent use in other examples. The parameter values used for this purpose are the deterministic parameter values shown in Example 1.1 and the following noise parameter values:

- $\sigma_{11} = \sigma_{22} = \sigma_{33} = 0$, $S_{11} = 0.01$, $S_{22} = 0.001$, $S_{33} = 0.01$ (Figure 2.1).
- $\sigma_{11} = \sigma_{22} = \sigma_{33} = 0.1$, $S_{11} = 0.01$, $S_{22} = 0.001$, $S_{33} = 0.01$ (Figure 2.2).
- $\sigma_{11} = \sigma_{22} = \sigma_{33} = 0.3162$, $S_{11} = 0.01$, $S_{22} = 0.001$, $S_{33} = 0.01$ (Figure 2.3).

A discretization time interval corresponding to $T = 10000$ is used and every 100'th value is sampled to give data sets containing 101 samples each ($N = 101$). ■



(a) Batch no. 1. (b) Batch no. 2.

Figure 2.1. Batch data sets generated in Example 2.2 - first noise parameter set. Solid staircase: Feed flow rate F; dashed lines: Biomass measurements y_1; dotted lines: Substrate measurements y_2; dash-dotted lines: Volume measurements y_3.

2.1.4 Filtering theory

As shown by Jazwinski (1970), Itô SDE's provide the basis for continuous-discrete nonlinear filtering, which is an important topic within the proposed grey-box modelling framework, because it involves determining estimates of the state variables of a continuous time system from noisy discrete time observations of the output variables. More specifically, the general continuous-discrete nonlinear filtering problem is based on a model of the following type:

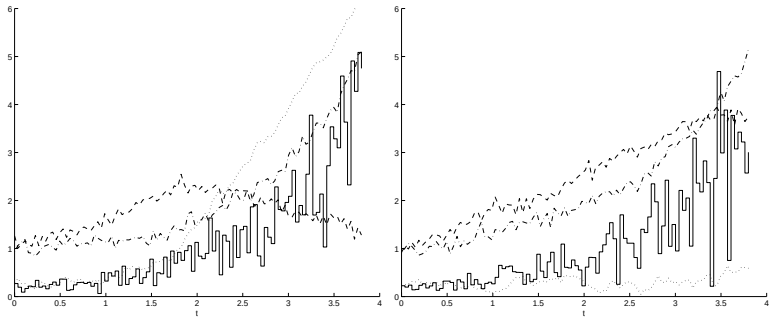
$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, t)dt + \boldsymbol{\sigma}(\boldsymbol{x}_t, t)d\boldsymbol{\omega}_t \ , \ t \geq 0 \tag{2.27}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, t_k) + \boldsymbol{e}_k \ , \ k = 0, 1, \ldots \tag{2.28}$$

where $\boldsymbol{x}_t \in \mathbb{R}^n$ is a state vector, $\boldsymbol{y}_k \in \mathbb{R}^l$ is an output vector, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process, $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}_k)$ and $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ and $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ are nonlinear functions. If these functions satisfy appropriate conditions (Jazwinski, 1970), the Itô solution $\{\boldsymbol{x}_t\}$ to the system equation of the model is a Markov process and can be characterized by its probability density $p(\boldsymbol{x}_t)$, $t \geq 0$, the evolution of which can be determined by solving the equation:

$$\frac{\partial p}{\partial t} = -\sum_{i=1}^{n} \frac{\partial (p f_i)}{\partial x_i} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 (p(\boldsymbol{\sigma}\boldsymbol{\sigma}^T)_{ij})}{\partial x_i \partial x_j} \tag{2.29}$$

for $t \geq 0$ with initial condition $p(\boldsymbol{x}_0)$. Here p is shorthand for $p(\boldsymbol{x}_t)$, f_i is the i'th element of $\boldsymbol{f}(\cdot)$ and $(\boldsymbol{\sigma}\boldsymbol{\sigma}^T)_{ij}$ is the ij-element of $\boldsymbol{\sigma}(\cdot)\boldsymbol{\sigma}(\cdot)^T$. This equation is known as *Kolmogorov's forward equation* or the *Fokker-Planck equation* and



(a) Batch no. 1. (b) Batch no. 2.

Figure 2.2. Batch data sets generated in Example 2.2 - second noise parameter set. Solid staircase: Feed flow rate F; dashed lines: Biomass measurements y_1; dotted lines: Substrate measurements y_2; dash-dotted lines: Volume measurements y_3.

is one of the two essential equations of continuous-discrete nonlinear filtering, because it can also be used to describe the evolution between observations of the probability density of interest for this problem, i.e.:

$$p(\boldsymbol{x}_t|\mathcal{Y}_k) = p(\boldsymbol{x}_t|\boldsymbol{y}_k, \boldsymbol{y}_{k-1}, \ldots, \boldsymbol{y}_1, \boldsymbol{y}_0) \ , \ t \in [t_k, t_{k+1}] \tag{2.30}$$

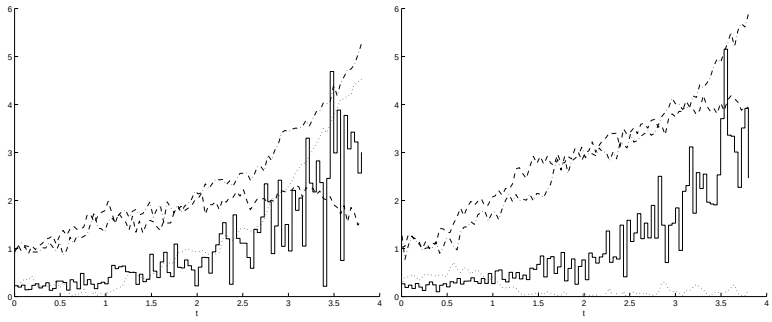
which is the conditional probability density of $\boldsymbol{x}_t$ given all observations available at time t_k. The other essential equation of continuous-discrete nonlinear filtering describes how the conditional probability density changes when a new observation $\boldsymbol{y}_{k+1}$ is obtained and is based on *Bayes' rule*:

$$p(\boldsymbol{x}_{k+1}|\mathcal{Y}_{k+1}) = \frac{p(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1})p(\boldsymbol{x}_{k+1}|\mathcal{Y}_k)}{\int p(\boldsymbol{y}_{k+1}|\boldsymbol{\xi})p(\boldsymbol{\xi}|\mathcal{Y}_k)d\boldsymbol{\xi}} \tag{2.31}$$

where $\int p(\boldsymbol{y}_{k+1}|\boldsymbol{\xi})p(\boldsymbol{\xi}|\mathcal{Y}_k)d\boldsymbol{\xi}$ is simply $p(\boldsymbol{y}_{k+1}|\mathcal{Y}_k)$ and:

$$p(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1}) = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\varepsilon}_{k+1}^T \boldsymbol{S}_k^{-1} \boldsymbol{\varepsilon}_{k+1})\right)}{\sqrt{\det(\boldsymbol{S}_k)}\left(\sqrt{2\pi}\right)^l} \tag{2.32}$$

where $\boldsymbol{\varepsilon}_{k+1} = \boldsymbol{y}_{k+1} - \boldsymbol{h}(\boldsymbol{x}_{k+1}, t_{k+1})$. Altogether (2.29) and (2.31) provide the analytical framework for solving the general continuous-discrete nonlinear filtering problem in terms of probability densities. However, (2.29) can only be solved explicitly in very simple cases, and numerical solution of this equation is computationally prohibitive. Furthermore, a solution in terms of e.g. first and second order moments is often more useful for practical purposes. As shown by Jazwinski (1970), an analytical framework for obtaining a solution of this



(a) Batch no. 1. (b) Batch no. 2.

Figure 2.3. Batch data sets generated in Example 2.2 - third noise parameter set. Solid staircase: Feed flow rate F; dashed lines: Biomass measurements y_1; dotted lines: Substrate measurements y_2; dash-dotted lines: Volume measurements y_3.

type can also be established. Unfortunately, this solution is seldom computationally realizable either, because it depends on higher order moments as well (Jazwinski, 1970). In the general case, approximations are therefore needed to obtain a realizable filtering solution. A number of such approximations are available (Jazwinski, 1970; Maybeck, 1982), one of which is the extended Kalman filter (EKF), which is applied within the parameter estimation method of the proposed grey-box modelling framework (see Section 2.2). The EKF is based on the ordinary Kalman filter, which, if the diffusion term is independent of the state variables, provides an exact solution to the filtering problem for linear systems, i.e. systems where the system equation consists of a set of linear SDE's and the measurement equation is also linear in the state variables.

2.1.5 Stochastic control theory

As shown by Åström (1970) models of the type (2.27)-(2.28), with additional manipulable input variables, also provide the basis for stochastic optimal control with simultaneous state estimation. Approximate methods are also needed to solve this problem in the general case, and only for linear systems, where the separation theorem applies, an exact closed-form solution is available.

Developing specific methods for optimal control with simultaneous state estimation is outside the scope of the work presented in this thesis and the topic has merely been mentioned here to illustrate the power of continuous-discrete stochastic state space models in terms of also facilitating such developments.

2.2 Parameter estimation

The second element of the grey-box modelling cycle deals with estimation of the unknown parameters of the continuous-discrete stochastic state space model in (2.2)-(2.3) from experimental data. This is not only important in order to find appropriate parameter values for the physically meaningful parameters occuring in the drift term of the system equation, but also in order to assess the uncertainty of the resulting model, which can be done by evaluating the statistical significance of the parameters of the corresponding diffusion term based on estimates of these. In particular, if a diagonal parameterization of the diffusion term is used, estimation of the parameters of this term facilitates pinpointing of model deficiencies as shown in Section 2.5. A parameter estimation method is therefore needed, which allows simultaneous estimation of all unknown parameters occuring in (2.2)-(2.3) based on experimental data.

Given the nature of fed-batch processes, which is reflected by the model in (2.2)-(2.3), the estimation method must be able to handle nonlinear discretely, partially observed systems with measurement noise and it must be applicable to relatively large multivariate systems. Furthermore, it must be able to provide a measure of the uncertainty of the individual parameter estimates in order to facilitate subsequent application of statistical tests. Provided these primary requirements are fulfilled, secondary requirements for the estimation method are computational efficiency and ease of use. Finally, because several sets of experimental data from separate batch runs are often available, a method that allows use of multiple independent data sets for the estimation is preferred.

2.2.1 Maximum likelihood estimation

The properties of the model in (2.2)-(2.3) facilitate application of a probabilistic estimation method such as *maximum likelihood* (ML). Given the observations:

$$\mathcal{Y}_N = [\boldsymbol{y}_N, \ldots, \boldsymbol{y}_k, \ldots, \boldsymbol{y}_1, \boldsymbol{y}_0] \tag{2.33}$$

ML estimates of the unknown parameters can be determined by finding the parameters $\boldsymbol{\theta}$ that maximize the likelihood function, i.e.:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = p(\mathcal{Y}_N|\boldsymbol{\theta}) \tag{2.34}$$

which is simply the joint probability density of the observations $\mathcal{Y}_N$ given the parameters $\boldsymbol{\theta}$. The likelihood function can also be written as follows:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} p(\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}) \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{2.35}$$

where the rule $P(A \cap B) = P(A|B)P(B)$ has been applied to form a product of conditional probability densities. Given the initial probability density $p(\boldsymbol{y}_0|\boldsymbol{\theta})$,

all subsequent conditional densities and hence the likelihood function can be determined by solving a continuous-discrete nonlinear filtering problem, as shown in Section 2.1. The parameter estimates can then be determined by maximizing the likelihood function, e.g. by solving the optimisation problem:

$$\min_{\boldsymbol{\theta}\in\Theta} \{-\ln\left(L(\boldsymbol{\theta};\mathcal{Y}_N)\right)\} \tag{2.36}$$

or the corresponding estimating equation:

$$\boldsymbol{S}_N(\boldsymbol{\theta};\mathcal{Y}_N) = \frac{d\ln\left(L(\boldsymbol{\theta};\mathcal{Y}_N)\right)}{d\boldsymbol{\theta}} = \mathbf{0} \tag{2.37}$$

but, unfortunately, neither approach is feasible in the general case, because solving the continuous-discrete nonlinear filtering problem is computationally prohibitive, and an alternative estimation method is therefore needed.

In a recent review paper, Nielsen *et al.* (2000a) have considered a number of different parameter estimation methods for nonlinear discretely observed Itô SDE's, which all provide alternatives to the ML method described above, either in terms of approximations or in terms of alternative formulations of the problem. In the following a brief outline of these methods is given, and they are evaluated in terms of their applicability for estimation of the unknown parameters of the model in (2.2)-(2.3), before a specific method is selected.

2.2.2 Likelihood-based methods

The first group of methods considered by Nielsen *et al.* (2000a) are *likelihood-based methods*, which are methods that seek to approximate the ML method described above. In one method this is done by discretizing a likelihood function obtained by assuming that continuous observations are available, and in another method it is done by computing the likelihood function for a discretized version of the model. Neither of these methods apply to partially observed systems nor allow measurement noise, however, and the former does not allow estimation of the parameters of the diffusion term either. A somewhat more powerful likelihood-based method, which applies to partially observed systems, is a method based on Markov Chain Monte Carlo (MCMC) methodology, but, unfortunately, this method does not allow measurement noise either.

2.2.3 Methods of moments

Another group of methods considered by Nielsen *et al.* (2000a) are *methods of moments*, where parameter estimates are obtained by matching certain moment conditions for a discretized version of the model. These methods are less computationally demanding than likelihood-based methods, because they are based on moment conditions instead of complete probability densities. A number of different methods of moments are available, e.g. the Generalized Method

of Moments (GMM), which, however, does not apply to partially observed systems nor allow measurement noise. The Efficient Method of Moments (EMM) and the Indirect Inference (II) method are both extensions of the GMM, which apply to partially observed systems but do not allow measurement noise either.

2.2.4 Estimating functions

A group of estimation methods that may be seen as an intermediate between likelihood-based methods and methods of moments are *estimating functions*, an introduction to the application of which for purposes not related to SDE's is given by Heyde (1997). In this context estimating functions provide a very general framework for estimation, as it can be shown that this methodology encompasses ML (under certain conditions), least squares (LS), weighted least squares (WLS) and a number of other methods. The idea of estimating functions is to choose an appropriate function $\boldsymbol{G}_N(\cdot) \in \mathbb{R}^r$, $r \in \mathbb{N}$, of the observations $\mathcal{Y}_N$ and the unknown parameters $\boldsymbol{\theta}$, which satisfies the estimating equation:

$$\boldsymbol{G}_N(\boldsymbol{\theta}; \mathcal{Y}_N) = \mathbf{0} \tag{2.38}$$

and solve this equation for $\boldsymbol{\theta}$. An example of an estimating function is $\boldsymbol{S}_N(\cdot)$ in (2.37), which, because it is the derivative of the logarithm of the likelihood function, is based on complete probability densities, but estimating functions need in fact only be a function of certain moments. In particular, an estimating function of the so-called linear family, which can be viewed as a first order Taylor expansion of $\boldsymbol{S}_N(\cdot)$, only requires first and second order moments, whereas an estimating function of the so-called quadratic family (equivalent to a second order Taylor expansion of $\boldsymbol{S}_N(\cdot)$) requires higher order moments as well. A major advantage of estimating functions is that precise mathematical statements about how to choose these functions in an optimal way can be made by maximizing the so-called Godambe information (Heyde, 1997), which provides an optimal trade-off between bias and variance for the resulting estimator.

In the context of parameter estimation for nonlinear discretely observed SDE's, a number of methods based on estimating functions have been proposed, e.g. the Martingale Estimating Functions (MEF's) by Bibby and Sørensen (1995), which are estimating functions of the linear family based on first and second order conditional moments. These MEF's do not allow estimation of the parameters of the diffusion term, but with the MEF's proposed by Bibby and Sørensen (1996), which are of the quadratic family and based on higher order conditional moments as well, this is possible. Unfortunately, neither type of MEF's apply to partially observed systems nor allow measurement noise.

The Prediction-Based Estimating Functions (PEF's) proposed by Sørensen (1999), which are based on unconditional instead of conditional moments, provide a way of handling partially observed systems but still do not allow measurement noise. Nielsen *et al.* (2000b) have recently proposed an extension

of the PEF's to handle measurement noise, and, in principle, these Prediction-Based Estimating Functions with Measurement noise (PEFM's) are sufficiently general to be applicable for estimation of the unknown parameters of the model in (2.2)-(2.3). Unfortunately, the PEFM's require that the measurement equation of the model can be expressed in terms of polynomials, which is not always the case, and they are based on a large number of unconditional moments, the determination of which easily becomes computationally prohibitive.

2.2.5 Filtering-based methods

A group of methods with greater application potential for estimation of the unknown parameters of the model in (2.2)-(2.3) are *filtering-based methods*, which seek to approximate the ML method described above by incorporating computationally realizable approximate solutions to the continuous-discrete nonlinear filtering problem. In the general case, higher-order filters (Maybeck, 1982) are needed, but since the diffusion term has been assumed to be independent of the state variables, an approximation based on the EKF (Jazwinski, 1970) can be applied. More specifically, since the SDE's of the model are driven by a Wiener process, and since increments of a Wiener process are Gaussian, it is reasonable to assume that the conditional probability densities constituting the likelihood function can be well approximated by Gaussian densities, which means that the EKF can be applied. Using this argument, an estimation method incorporating the EKF has been proposed by Madsen and Melgaard (1991) and Melgaard and Madsen (1993), where, because the Gaussian density is completely characterized by its mean and covariance, the likelihood function becomes:

$$L(\boldsymbol{\theta};\mathcal{Y}_N) = \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1}\boldsymbol{\epsilon}_k\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{2.39}$$

where $\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1}$, $\hat{\boldsymbol{y}}_{k|k-1} = E\{\boldsymbol{y}_k|\mathcal{Y}_{k-1},\boldsymbol{\theta}\}$ and $\boldsymbol{R}_{k|k-1} = V\{\boldsymbol{y}_k|\mathcal{Y}_{k-1},\boldsymbol{\theta}\}$ can be computed recursively by means of the EKF. The assumption of Gaussianity is only likely to hold for small sample times, and the validity of this assumption should therefore be checked subsequent to the estimation, but as shown in Section 2.3 this is straightforward, because several tools are available for this purpose. An additional benefit of the EKF-based method by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) is that, if prior information about the parameters is available in the form of a prior Gaussian probability density function $p(\boldsymbol{\theta})$, Bayes' rule can be applied to give an improved estimate by forming the posterior probability density function:

$$p(\boldsymbol{\theta}|\mathcal{Y}_N) = \frac{p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{Y}_N)} \propto p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{2.40}$$

and subsequently finding the parameters that maximize this function, i.e. by performing *maximum a posteriori* (MAP) estimation. Altogether, the EKF-

based method fulfills the primary requirements stated in the beginning of this section, because it is able to handle nonlinear discretely, partially observed systems with measurement noise and applies to relatively large multivariate systems, and because it provides a measure of the uncertainty of the individual parameter estimates. Therefore this method has been selected for the parameter estimation part of the proposed grey-box modelling framework.

2.2.6 Implementation of the EKF-based method

As a part of the work presented in this thesis, the EKF-based method has been further developed to make it more readily applicable for estimation of the unknown parameters of the model in (2.2)-(2.3). In particular, because the original method was unable to handle models with singular Jacobians, which are very common in the context of fed-batch process modelling, an alternative solution based on the singular value decomposition (SVD) has been developed, and the method has been extended to allow the use of multiple independent sets of experimental data for the estimation and to handle missing observations in a much more appropriate way. The details of these developments are given in Appendix A, which provides a complete mathematical outline of the algorithms of the computer program **CTSM**, within which the extended method has been implemented. **CTSM**, which is based on a similar computer program by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) called CTLSM, has been equipped with a graphical user interface for ease of use, and to increase the computational efficiency the binary code has been optimized and prepared for shared memory parallel computing, as shown in Appendix A.

As discussed in Chapter 1, the use of continuous-discrete stochastic state space models such as (2.2)-(2.3) facilitates estimation of unknown parameters in a PE setting, which is generally more advantageous than estimation in an OE setting. To illustrate this, a comparison between the method implemented in **CTSM**, which is a PE estimation method, and a conventional OE estimation method is given in Chapter 3. Furthermore, a comparison between **CTSM** and a computer program implementing a similar estimation method by Bohlin and Graebe (1995) and Bohlin (2001) is given in the paper included in Appendix D. The purpose of this comparison has been to reveal some very important differences between the two methods, which render the program by Bohlin and Graebe (1995) and Bohlin (2001) inappropriate for estimation of the parameters of the diffusion term and hence for application within the proposed grey-box modelling framework. To illustrate the use of parameter estimation in the context of this framework, a simple example is given in the following.

Example 2.3 (Parameter estimation for the fermentation process model)
This example illustrates the use of parameter estimation in the context of the proposed grey-box modelling framework using a variant of the re-formulated model of the fermentation process shown in Example 2.1 and data from Example 2.2. To illustrate the possibility of using the proposed grey-box modelling framework for systematic

iterative model improvement, it is assumed from now on that the true structure of the growth rate is unknown, and $\mu(S)$ is therefore replaced by a constant μ to yield a preliminary model with the following system equation:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -\frac{\mu X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \ , \ t \in [t_0, t_f] \tag{2.41}$$

and the following measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{2.42}$$

Using **CTSM** and the data set shown in Figure 2.1a, the estimates (and standard deviations and t-scores) shown in Table 2.1 are obtained for this model. These results will be used in subsequent examples, so further discussion is postponed. ■

2.3 Residual analysis

The third element of the grey-box modelling cycle deals with obtaining information about the quality of the continuous-discrete stochastic state space model in (2.2)-(2.3), once the unknown parameters have been estimated. An important aspect in this regard is to investigate the prediction capabilities of the model over a prediction horizon appropriate for its intended purpose, which can be done by performing cross-validation and examining the corresponding residuals. Residual analysis can be performed in a one-step-ahead prediction setting (based on $\hat{\boldsymbol{y}}_{k|k-1}$) as well as a pure simulation setting (based on $\hat{\boldsymbol{y}}_{k|0}$), and, depending on the intended purpose of the model, one may be more appropriate than the other. In the context of the proposed grey-box modelling framework, however, the pure simulation setting is the most important, as the

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.6973E-01	3.4150E-02	28.3962	Yes
S_0	2.5155E-01	3.1938E-02	7.8761	Yes
V_0	1.0384E+00	1.8238E-02	56.9359	Yes
μ	6.8548E-01	2.2932E-02	29.8921	Yes
σ_{11}	1.8411E-01	2.5570E-02	7.2000	Yes
σ_{22}	2.2206E-01	3.4209E-02	6.4912	Yes
σ_{33}	2.7979E-02	1.7943E-02	1.5594	No
S_{11}	6.7468E-03	1.3888E-03	4.8580	Yes
S_{22}	3.9131E-04	2.4722E-04	1.5828	No
S_{33}	1.0884E-02	1.5409E-03	7.0633	Yes

Table 2.1. Estimation results. Model in (2.41)-(2.42) - data from Figure 2.1a.

models being developed must be applicable for subsequent state estimation and optimal control, where the latter requires models with good long-term prediction capabilities. This is discussed in more detail in Section 2.4.

As shown in Appendix A, **CTSM** facilitates residual analysis in both settings by allowing predictions ($\hat{\boldsymbol{y}}_{k|k-1},\ k = 0, \ldots, N$, and $\hat{\boldsymbol{y}}_{k|0},\ k = 0, \ldots, N$) to be computed for a given set of cross-validation data by means of the EKF.

2.3.1 Performing residual analysis

The idea of residual analysis more specifically is to determine if the residuals can be regarded as white noise, and a number of different methods can be applied for this purpose (Brockwell and Davis, 1991; Holst *et al.*, 1992).

For linear systems, one of the most powerful of these methods is to compute and inspect the standard correlation functions, i.e. the *sample autocorrelation function* (SACF) and the *sample partial autocorrelation function* (SPACF) of the residuals and the *sample cross-correlation function* (SCCF) between the residuals and the inputs, to detect if there are any significant lag dependencies, as this indicates that the residuals cannot be regarded as white noise. More details about the standard correlation functions are given in Appendix B.

For nonlinear systems, extensions of these functions have been proposed by Nielsen and Madsen (2001a) in the form of the *lag dependence function* (LDF), the *partial lag dependence function* (PLDF), the *crossed lag dependence function* (CLDF) and the *nonlinear lag dependence function* (NLDF), which are all based on a close relation between correlation coefficients and the coefficients of determination for regression models and extend to nonlinear systems by incorporating various nonparametric regression models. Unlike the standard correlation functions, these functions can also detect certain nonlinear dependencies and are therefore extremely useful for residual analysis within the proposed grey-box modelling framework. More details about these functions are given in Appendix B, and the following simple example illustrates their use.

Example 2.4 (Residual analysis for the fermentation process model)
This example illustrates the use of residual analysis for the preliminary fermentation process model shown in Example 2.3 subsequent to estimating the parameters. Figure 2.4 shows cross-validation residual analysis results obtained using the data set shown in Figure 2.1b. These results show that the pure simulation capabilities of the model are poor, whereas its one-step-ahead prediction capabilities are quite good. ■

As mentioned in Section 2.2 the Gaussianity assumption inherent to the EKF-based parameter estimation method is only likely to hold for small sample times and should be checked subsequent to the estimation. A number of tools are available for this purpose (Holst *et al.*, 1992; Bak *et al.*, 1999), including the above residual analysis tools. If, by applying these tools to residuals obtained in a one-step-ahead prediction setting from the estimation data set, there are

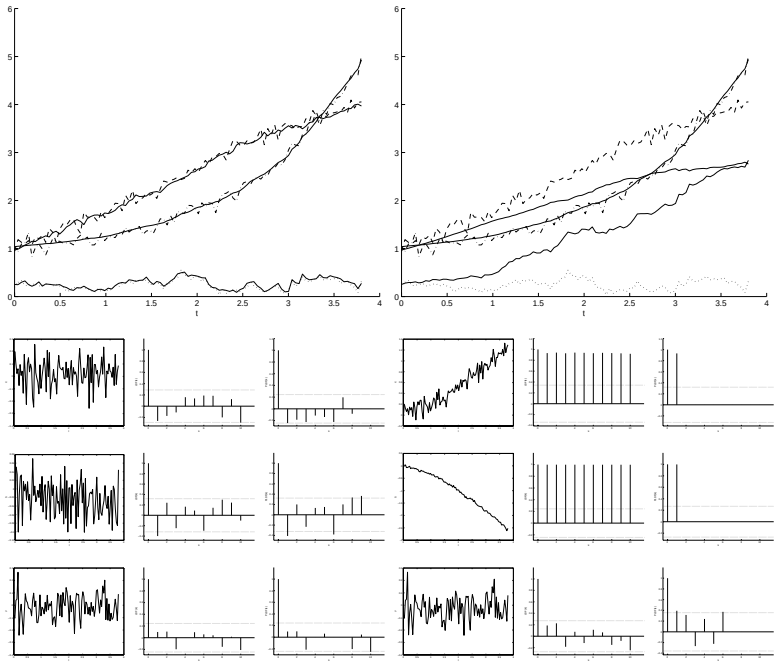


Figure 2.4. Cross-validation residual analysis results for the model in Example 2.3 with parameters in Table 2.1 using the validation data set shown in Figure 2.1b. Top left: One-step-ahead prediction comparison (solid lines: Predicted values); top right: Pure simulation comparison (solid lines: Simulated values); bottom left: One-step-ahead prediction residuals, LDF and PLDF for y_1, y_2 and y_3; bottom right: Pure simulation residuals, LDF and PLDF for y_1, y_2 and y_3.

no significant lag dependencies, this is an indication that the residuals can be regarded as white noise and hence that the assumption is valid. If this is the case, the statistical tests described in Section 2.5 can also be applied at this point to provide information about the quality of the model in (2.2)-(2.3).

More specifically, it can be determined if some of the parameters of the model are insignificant, indicating that the model is overly complex and that these parameters may be eliminated. In practice, however, the Gaussianity assumption is only likely to be valid if the structure of the model is appropriate, which means that these tests should only be applied in the final stages of model development. As discussed in much more detail in Section 2.5, applying these tests to the parameters of the diffusion term nevertheless provides reasonable indications, facilitating pinpointing of model deficiencies in early stages as well.

2.4 Model falsification or unfalsification

The fourth element of the grey-box modelling cycle deals with determining whether or not, based on the information about its quality obtained by performing residual analysis, the model in (2.2)-(2.3) is sufficiently accurate to be applied for state estimation and optimal control. If this is the case, the model is said to be *unfalsified* with respect to the available information, and the model development procedure implied by the grey-box modelling cycle can be terminated. If not, the model is said to be *falsified*, and the model development procedure must be repeated by returning to the model (re)formulation element of the grey-box modelling cycle and altering the model in an appropriate way.

2.4.1 Evaluating model quality

In order to evaluate whether or not the model in (2.2)-(2.3) is sufficiently accurate to be applied for state estimation and optimal control, an evaluation of its prediction capabilities is essential. However, the specific degree of accuracy required is essentially an application-specific and therefore often subjective measure, which means that, in general, this evaluation cannot be based on a specific test. Ultimately, i.e. to achieve the highest possible degree of accuracy, a test for whiteness of cross-validation residuals obtained in a pure simulation setting can be used, because good long-term prediction capabilities are essential for optimal control of fed-batch processes. More specifically, although developing methods for optimal control with simultaneous state estimation is outside the scope of the work presented in this thesis, it is evident that for a model of the type in (2.2)-(2.3) to be applicable for e.g. MPC, it must be able to predict the future evolution of the system over wide ranges of state space, because this methodology relies on long-term prediction. This also implies that, ideally, none of the parameters of the diffusion term should be significant either, because this means that significant parts of the variation in the experimental data cannot be explained by the corresponding drift term, which it must if e.g. MPC is to be applied, unless an alternative implementation is developed, which takes the uncertainty implied by a significant diffusion term into account. In any case, the model should not be overly complex either, so if the model has insignificant parameters, it should be considered to eliminate some of them.

Example 2.5 (Evaluating the quality of the fermentation process model)
This example illustrates the procedure for evaluating model quality for the preliminary fermentation process model shown in Example 2.3 subsequent to estimating the parameters. The residual analysis results obtained in Example 2.4 show that the pure simulation capabilities of the model are poor by indicating that the corresponding residuals cannot be regarded as white noise. This means that the model cannot be applied for state estimation and optimal control, because good long-term prediction capabilities are needed for the latter. Hence the model is falsified. ■

2.5 Statistical tests

The fifth element of the grey-box modelling cycle deals with detecting and pinpointing deficiencies in the model in (2.2)-(2.3), if, based on the above evaluation of its quality, the model is falsified for the purpose of state estimation and optimal control, and, as it turns out, the particular nature of the model facilitates this task. More specifically, statistical tests for significance of the individual parameters, particularly the parameters of the diffusion term, can be applied. However, if the residual sequences obtained in the residual analysis element of the grey-box modelling cycle can be regarded as stationary time series, the residual analysis tools mentioned in Section 2.3 can also be applied at this stage. More specifically, like the standard correlation functions, the nonlinear extensions of these functions can be applied for structural identification, e.g. to determine if more state variables are needed. A more elaborate discussion of this particular topic is given by Nielsen and Madsen (2001a).

Applying statistical tests to determine the significance of individual parameters is generally important in terms of investigating if the structure of a model is appropriate. In principle, *insignificant* parameters are parameters that may be eliminated, and the presence of such parameters is therefore an indication that the model is overly complex. On the other hand, because of the particular nature of the model in (2.2)-(2.3), where the diffusion term is included to account for uncertainty, the presence of *significant* parameters in this term is an indication that the corresponding drift term is unable to explain significant parts of the variation in the experimental data. This provides a measure that allows model deficiencies to be detected. If a diagonal parameterization of the diffusion term has been used, this even allows the deficiencies to be pinpointed in the sense that deficiencies in specific elements of the drift term can be detected.

In terms of a specific test methodology, it is shown in Appendix A that, by the central limit theorem, the EKF-based parameter estimation method discussed in Section 2.2 provides parameter estimates that are asymptotically Gaussian, and that it also provides an estimate of the corresponding covariance matrix, on the basis of which tests for insignificance can be performed. In particular, marginal t-tests can be performed to test the following hypothesis:

$$H_0: \quad \theta_j = 0 \tag{2.43}$$

against the corresponding alternative:

$$H_1: \quad \theta_j \neq 0 \tag{2.44}$$

i.e. to test whether a specific parameter θ_j is insignificant or not. The test quantity is the value of the parameter estimate divided by its standard deviation, and under H_0 this quantity is asymptotically t-distributed with a number of degrees of freedom that equals the total number of observations minus the number of parameters that have been estimated. More details about this test are given in Appendix B, and the following is a simple example of its use.

Example 2.6 (Marginal t-tests for the fermentation process model)
This example illustrates the use of marginal t-test for parameter insignificance for the preliminary fermentation process model shown in Example 2.3 subsequent to obtaining the estimation results shown in Table 2.1. This table also includes t-scores computed from the estimates and their standard deviations, indicating that, on a 5% level, only one of the parameters of the diffusion term is insignificant, i.e. σ_{33}. That σ_{11} and σ_{22} are both significant is an indication that there is significant variation in the experimental data, which cannot be explained by the corresponding elements of the drift term, in turn indicating that these elements may be deficient. ■

Due to correlations between the individual parameter estimates, a series of marginal tests of the above type cannot be used to test the hypothesis that a subset of the parameters $\boldsymbol{\theta}_* \subset \boldsymbol{\theta}$ are simultaneously insignificant:

$$H_0: \quad \boldsymbol{\theta}_* = \mathbf{0} \tag{2.45}$$

against the alternative that they are not:

$$H_1: \quad \boldsymbol{\theta}_* \neq \mathbf{0} \tag{2.46}$$

Hence a test that takes correlations into account must be used instead, e.g. a likelihood ratio test, a Lagrange multiplier test or a test based on Wald's W-statistic (Holst *et al.*, 1992). Under H_0 the test quantities for these tests all have the same asymptotic χ^2-distribution with a number of degrees of freedom that equals the number of parameters subjected to the test (Holst *et al.*, 1992), but in the context of the proposed grey-box modelling framework the test based on Wald's W-statistic has the advantage that no re-estimation of the parameters is required. More details about this test are also given in Appendix B.

Strictly speaking, the above tests should only be applied if the Gaussianity assumption mentioned in Section 2.2 is valid, which is only likely to be the case in the final stages of model development, where the structure of the model is appropriate, as discussed in Section 2.3. Nevertheless, the corresponding test results can be used to provide reasonable indications in early stages as well.

2.5.1 Pinpointing model deficiencies

If a diagonal parameterization of the diffusion term of the model in (2.2)-(2.3) has been used, the measure mentioned above for detecting model deficiencies can be used to pinpoint these deficiencies as well, in the sense that deficiencies in specific elements of the drift term can be detected. More specifically, the presence of significant parameters in a given diagonal element of the diffusion term is an indication that the corresponding element of the drift term may be deficient, in turn suggesting that some of the phenomena occuring in this term may be inappropriately modelled. With this information at hand, it may be possible, by using physical insights, to subsequently select a specific suspect phenomenon for further investigation, whereupon the proposed grey-box modelling framework provides means to confirm if this suspicion is true.

More specifically, suspect phenomena are typically reaction rates, heat and mass transfer rates and similar complex dynamic phenomena, all of which can usually be described using functions of the state variables, i.e.:

$$r_t = \varphi(\boldsymbol{x}_t, \boldsymbol{\theta}) \tag{2.47}$$

where r_t symbolizes the phenomenon of interest and $\varphi(\cdot) \in \mathbb{R}$ is the nonlinear function used to describe it. This means that the suspicion that $\varphi(\cdot)$ is inappropriate can be confirmed by estimating the parameters of a re-formulated version of the model and performing statistical tests to determine the significance of the parameters of the diffusion term of this model. In the re-formulated version of the model r_t is included as an additional state variable as follows:

$$d\boldsymbol{x}_t^* = (\boldsymbol{f}^*(\boldsymbol{x}_t^*, t, \boldsymbol{\theta}) + \boldsymbol{g}^*(\boldsymbol{x}_t^*, t, \boldsymbol{\theta})\boldsymbol{u}_t)dt + \boldsymbol{\sigma}^*(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t^* \tag{2.48}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k^*, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{2.49}$$

where $\boldsymbol{x}_t^* = [\boldsymbol{x}_t^T \; r_t]^T$ is an augmented state vector, $\boldsymbol{\sigma}^*(\cdot) \in \mathbb{R}^{(n+1)\times(n+1)}$ is a nonlinear function, $\{\boldsymbol{\omega}_t^*\}$ is an $(n+1)$-dimensional standard Wiener process and $\boldsymbol{f}^*(\cdot) \in \mathbb{R}^{n+1}$ and $\boldsymbol{g}^*(\cdot) \in \mathbb{R}^{(n+1)\times m}$ are functions defined as follows:

$$\boldsymbol{f}^*(\boldsymbol{x}_t^*, t, \boldsymbol{\theta}) = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}_t, t, \boldsymbol{\theta}) \\ \frac{\partial \varphi(\boldsymbol{x}_t, \boldsymbol{\theta})}{\partial \boldsymbol{x}_t} \frac{d\boldsymbol{x}_t}{dt} \end{pmatrix} \tag{2.50}$$

$$\boldsymbol{g}^*(\boldsymbol{x}_t^*, t, \boldsymbol{\theta}) = \begin{pmatrix} \boldsymbol{g}(\boldsymbol{x}_t, t, \boldsymbol{\theta}) \\ \boldsymbol{0} \end{pmatrix} \tag{2.51}$$

If, upon estimating the unknown parameters of this model using a diagonal parameterization of the diffusion term, there are significant parameters in the particular diagonal element, which corresponds to r_t, this is a strong indication that $\varphi(\cdot)$ is in fact inappropriate and hence confirms the suspicion.

A particularly simple and very important special case of the above formulation is obtained if $\varphi(\cdot)$ has been assumed to be constant, in which case the partial derivative in (2.50) is zero and any variation in r_t must be explained by the corresponding diagonal element of the diffusion term. This in turn means that, if the parameters of this diagonal element are significant, this is an indication that $\varphi(\cdot)$ is not constant. This is illustrated in the following example.

Example 2.7 (Pinpointing deficiencies in the fermentation process model) This example illustrates the procedure for pinpointing model deficiencies for the preliminary fermentation process model shown in Example 2.3. The information obtained in Example 2.6 indicates that the first two elements of the drift term of this model may be deficient, and, since both of these elements depend on μ, this is a possible suspect for being deficient. To confirm this suspicion, the model is therefore re-formulated with μ as an additional state variable, which gives the following system equation:

$$d\begin{pmatrix} X \\ S \\ V \\ \mu \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -\frac{\mu X}{Y} + \frac{F(S_F - S)}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \;,\; t \in [t_0, t_f] \tag{2.52}$$

where, because μ has been assumed to be constant in Example 2.3, the last element of the drift term is zero. The measurement equation remains as in Example 2.3, i.e.:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{2.53}$$

Using **CTSM** and the same data set as in Example 2.3, the estimates (and standard deviations and t-scores) shown in Table 2.2 are obtained for this model. By performing marginal t-tests for parameter insignificance, it is revealed that, on a 5% level, only one of the parameters of the diffusion term is now significant, and because this is precisely the σ_{44} parameter corresponding to the equation for μ, the suspicion that μ is deficient is confirmed. More specifically, this is an indication that there is significant variation in μ and hence falsifies the constant assumption made in Example 2.3. ■

2.6 Nonparametric modelling

The sixth element of the grey-box modelling cycle deals with determining how to alter the model in (2.2)-(2.3) if it is falsified for the purpose of state estimation and optimal control and therefore needs to be improved by repeating the model development procedure implied by the grey-box modelling cycle. More specifically, the idea is to obtain nonparametric estimates of unknown functional relations and subsequently make inferences from these estimates to repair model deficiencies. The methods discussed in this section therefore require that specific model deficiencies have been pinpointed as shown in Section 2.5.

2.6.1 Estimating unknown functional relations

If a specific model deficiency has been pinpointed in the sense that it has been indicated that there is significant variation in the additional state variable r_t

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0239E+00	4.9566E-03	206.5723	Yes
S_0	2.3282E-01	1.1735E-02	19.8405	Yes
V_0	1.0099E+00	3.8148E-03	264.7290	Yes
μ_0	7.8658E-01	2.4653E-02	31.9061	Yes
σ_{11}	2.0791E-18	1.4367E-17	0.1447	No
σ_{22}	1.1811E-30	1.6162E-29	0.0731	No
σ_{33}	3.1429E-04	2.0546E-04	1.5297	No
σ_{44}	1.2276E-01	2.5751E-02	4.7674	Yes
S_{11}	7.5085E-03	9.9625E-04	7.5368	Yes
S_{22}	1.1743E-03	1.6803E-04	6.9887	Yes
S_{33}	1.1317E-02	1.3637E-03	8.2990	Yes

Table 2.2. Estimation results. Model in (2.52)-(2.53) - data from Figure 2.1a.

of the model in (2.48)-(2.49), which cannot be explained by the corresponding element of the drift term, this is a strong indication that the function $\varphi(\cdot)$ used to describe the phenomenon represented by r_t is inappropriate, i.e.:

$$\varphi(\boldsymbol{x}_t, \boldsymbol{\theta}) \neq \varphi_{\text{true}}(\boldsymbol{x}_t, \boldsymbol{\theta}) \tag{2.54}$$

where $\varphi_{\text{true}}(\cdot) \in \mathbb{R}$ is the "true" function. To repair this particular model deficiency a better estimate of $\varphi_{\text{true}}(\cdot)$ must therefore be obtained, i.e.:

$$\hat{\varphi}(\boldsymbol{x}_t, \boldsymbol{\theta}) \approx \varphi_{\text{true}}(\boldsymbol{x}_t, \boldsymbol{\theta}) \tag{2.55}$$

where $\hat{\varphi}(\cdot) \in \mathbb{R}$ is an appropriate function. As a first step towards obtaining a parametric expression for $\hat{\varphi}(\cdot)$ it turns out that a nonparametric estimate can be used. As shown in Appendix A, **CTSM** allows state estimates $\hat{\boldsymbol{x}}^*_{k|k}$, $k = 0, \ldots, N$, from the model in (2.48)-(2.49) to be computed for a given data set by means of the EKF. This means that a set of corresponding values of estimates of r_t and $\boldsymbol{x}_t$ can be obtained, provided that $\boldsymbol{x}^*_t$ is observable. On the basis of these values a nonparametric estimate of the functional relation between r_t and (a subset of) $\boldsymbol{x}_t$ can be obtained and plotted to visualize the structure of $\varphi_{\text{true}}(\cdot)$, and based on this visualization it may subsequently be possible to determine an appropriate parametric expression for $\hat{\varphi}(\cdot)$. Several univariate as well as multivariate nonparametric estimation methods are available (Hastie *et al.*, 2001). For univariate methods the problem is to obtain an estimate of the function $f(\cdot) \in \mathbb{R}$ in a model of the following type:

$$Y = f(X) + e \ , \ e \in N(0, \sigma^2) \tag{2.56}$$

based on a set of observations of a response variable Y and a single predictor variable X. Examples of such methods are piecewise polynomial smoothers, splines, kernel smoothers and wavelets, where the latter are well-suited for modelling discontinuities. Equivalently, the problem for multivariate methods is to estimate the function $f(\cdot) \in \mathbb{R}$ in a model of the following type:

$$Y = f(\boldsymbol{X}) + e \ , \ e \in N(0, \sigma^2) \tag{2.57}$$

based on a set of observations of a response variable Y and a vector $\boldsymbol{X}$ of several predictor variables $X_1, \ldots, X_p$. Examples of such methods are multidimensional splines, multidimensional kernel smoothers, additive models, regression trees, neural networks, Multivariate Adaptive Regression Splines (MARS) and Multiple Additive Regression Trees (MART). Of these, additive models are particularly simple, because they are based on the assumption that the contributions from the individual predictor variables are additive[4], i.e.:

$$Y = \alpha + \sum_{j=1}^{p} f_j(X_j) + e \ , \ e \in N(0, \sigma^2) \tag{2.58}$$

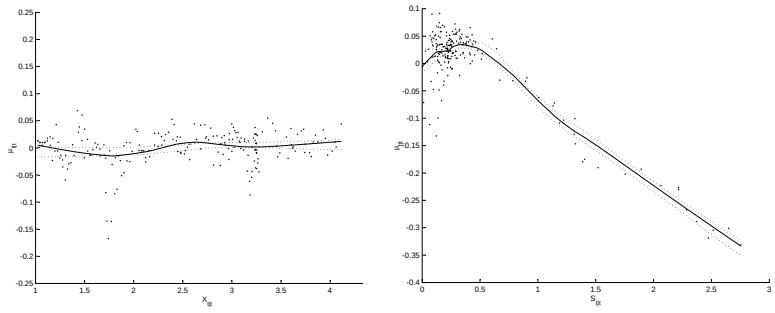
[4]The assumption of additive contributions does not necessarily limit the ability of additive models to provide estimates of non-additive functional relations, because functions of more than one predictor variable, e.g. $X_1 X_2$, can be included as predictor variables as well.

where α is a constant, which means that the contributions $f_j(\cdot) \in \mathbb{R}$, $j = 1, \ldots, p$, can be estimated separately by applying univariate methods in a recursive manner using the *backfitting algorithm* (Hastie and Tibshirani, 1990). Additive models also have the advantage of not suffering from the *curse of dimensionality*, which tends to render nonparametric estimation methods infeasible in higher dimensions. For this reason, and because the results obtained with such models are particularly easy to visualize by means of plots of estimates of the individual contributions $f_j(\cdot)$, $j = 1, \ldots, p$, with associated confidence intervals, additive models are preferred in the context of the proposed grey-box modelling framework. More specifically, since additive models may incorporate different univariate methods, additive models incorporating kernel smoothers are preferred, where the latter choice is due to the ease with which these can be implemented and to the fact that kernel smoothers only have one tuning parameter (the bandwidth) that must be selected. More details about kernel smoothers and additive models and related issues such as bandwidth optimisation and computation of bootstrap confidence intervals are given in Appendix C.

2.6.2 Making inferences from the estimates

Using additive models, the variation in r_t can be decomposed into the variation that can be attributed to each of (a subset of) the state variables (or each of a number of functions of more than one state variable) in turn, and the result can be visualized by means of plots of estimates of the individual contributions with associated confidence intervals. In this manner, it may be possible to reveal the structure of the "true" function $\varphi_{\text{true}}(\cdot)$ and get an idea how to formulate an appropriate parametric expression for an estimate $\hat{\varphi}(\cdot)$ of this function. In particular, it may be possible to determine which state variables have significant influence on the "true" function and which have not, and it may even be possible to determine how to model this influence with a parametric model. If the latter cannot be inferred directly from the nonparametric estimate by using physical insights, applying parametric curvefitting in a trial-and-error setting to find a good approximation to the nonparametric result is straightforward. In either case, valuable information can be obtained about how to alter the model in an appropriate way when the model development procedure is repeated. The use of nonparametric modelling is illustrated in the following simple example.

Example 2.8 (Improving the fermentation process model)
This example illustrates how nonparametric modelling can be used to determine how to alter the preliminary fermentation process model shown in Example 2.3 by repairing the model deficiency pinpointed in Example 2.7. The information obtained in Example 2.7 falsifies the assumption of constant μ made in Example 2.3, so to obtain a better estimate of the "true" function describing μ, state estimates $\hat{X}_{k|k}$, $\hat{S}_{k|k}$, $\hat{V}_{k|k}$ and $\hat{\mu}_{k|k}$, $k = 0, \ldots, N$, are computed from the model shown in Example 2.7 by using **CTSM** and the data sets shown in Figure 2.1, and by means of these an additive model can be fitted. It is reasonable to assume that μ does not depend on V, so only



(a) $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$. (b) $\hat{\mu}_{k|k}$ vs. $\hat{S}_{k|k}$.

Figure 2.5. Partial dependence plots of $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$ obtained by applying additive model fitting using locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths determined using 5-fold cross-validation). Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals computed from 1000 replicates (see Appendix C for details).

estimates of X and S are included in this model, which gives the results shown in Figure 2.5 in the form of partial dependence plots with associated bootstrap intervals.

From these plots it can be inferred that μ does not depend significantly on X (the estimate is almost constant over the range of X values), whereas there is a significant dependence on S (the estimate varies significantly over the range of S values). This result in turn suggests that the constant assumption made in Example 2.3 should be replaced with an assumption of μ being a function of S. More specifically, this function should comply with the functional relation revealed in Figure 2.5b. To a person with experience in fermentation process modelling, this functional relation is indicative of a growth rate that can be described by Monod kinetics with substrate inhibition (which is exactly the description used in Example 2.2 to generate the data sets mentioned above). In other words, a better (and in fact correct) estimate of the "true" function describing μ can be inferred directly in this particular case. ■

The above is an example of how, by fitting an additive model, a nonparametric estimate of the functional relation between r_t and (a subset of) the state variables can be obtained and visualized, and the example demonstrates that, based on this visualization, it can be determined that r_t depends on only one of the state variables in this case. The example also demonstrates how an appropriate parametric expression for this dependence can subsequently be inferred. However, due to correlation effects, the latter may not be equally straightforward if r_t depends on more than one of the state variables. More specifically, since additive models assume that the contributions from the individual predictor variables are additive, an actual dependence on e.g. the product between two

predictor variables or a fraction between them may be incorrectly interpreted as separate dependences on both of these variables, unless proper precautions are taken, e.g. by including the particular product or fraction as a predictor variable as well. Correlation effects and their implications are discussed in more detail in the application examples given in Chapter 3, which involve more complicated functional relations than the one in the above example. Based on experience gained from these application examples, some guidelines have been established to further systematize the use of nonparametric modelling in the context of the proposed grey-box modelling framework. They are given here:

1. Given a set of estimates of r_t and $\boldsymbol{x}_t$, start by excluding the variables in $\boldsymbol{x}_t$, which can be assumed not to influence r_t. Then fit an additive model of r_t vs. the remaining variables in $\boldsymbol{x}_t$, where these variables are included as single predictors, i.e. a simultaneous fit of Y vs. X_1, X_2, etc.

2. Based on this result, exclude the variables in $\boldsymbol{x}_t$, which do not seem to have any influence on r_t. If necessary, fit a new additive model of r_t vs. the remaining variables in $\boldsymbol{x}_t$, where these variables are again included as single predictors, i.e. a simultaneous fit of Y vs. X_1, X_2, etc.

3. Use this result to determine if r_t depends on more than one of the variables in $\boldsymbol{x}_t$. If so, fit new additive models, where, one at a time, products and fractions of these variables are included as predictors instead of the variables themselves, i.e. separate fits of Y vs. X_1X_2, $\frac{X_1}{X_2}$, $\frac{X_2}{X_1}$, etc.

Using these guidelines does not guarantee that sufficient information is obtained to make proper inferences about the "true" function describing r_t, but the application examples given in Chapter 3 have shown that these rules of thumb may be very useful in practice. In the third step, the separate inclusion of products and fractions instead of, and not along with, the variables themselves has been found necessary to ensure convergence of the backfitting algorithm.

2.7 Summary of the grey-box modelling cycle

The nonparametric modelling element described in Section 2.6 closes the loop shown in Figure 1.3 and thus completes the grey-box modelling cycle. As discussed in Section 1.3 the idea of the grey-box modelling cycle is to allow the quality of a model of a fed-batch process to be iteratively improved, until the model is unfalsified for the purpose of state estimation and optimal control with respect to the available information, or at least until no more information can be extracted from the available experimental data, in which case the model remains falsified until more experimental data becomes available. The methods behind the individual elements of the grey-box modelling cycle, which have been the focus of this chapter, facilitate this iterative procedure and can therefore be summarized in the form of an algorithm for systematic iterative model

improvement. This grey-box modelling algorithm has a number of key features, which make it very powerful in comparison with other approaches to grey-box modelling reported in literature, but it also has certain limitations. These key features and limitations are discussed after presenting the algorithm.

2.7.1 A grey-box modelling algorithm

Based on the individual elements of the grey-box modelling cycle, the following algorithm for systematic iterative model improvement for the purpose of state estimation and optimal control of fed-batch processes can be established:

1. Use first engineering principles and physical insights to derive an initial model structure in the form of an ODE model (see Section 2.1).
2. Translate the ODE model into a continuous-discrete stochastic state space model using a diagonal parameterization of the diffusion term to facilitate pinpointing of model deficiencies (see Section 2.1).
3. Estimate the unknown parameters of the model from experimental data with the EKF-based parameter estimation method (see Section 2.2).
4. Obtain information about the quality of the resulting model by performing cross-validation residual analysis (see Section 2.3).
5. Evaluate the obtained quality information to determine if the model is sufficiently accurate to be applied for subsequent state estimation and optimal control. If unfalsified, terminate model development. If falsified, proceed with model development (see Section 2.4).
6. Try to pinpoint specific model deficiencies by applying statistical tests and by re-formulating the model with additional state variables and repeating the estimation and test procedures (see Section 2.5).
7. If specific model deficiencies can be pinpointed, obtain state estimates from the re-formulated model and use additive models to obtain plots of appropriate estimates of functional relations (see Section 2.6).
8. Alter the model according to the estimated functional relations combined with physical insights and repeat from Step 3 (see Section 2.6).

The basic idea behind this grey-box modelling algorithm is to iteratively improve the quality of the model by systematically pinpointing and repairing model deficiencies, until a model is obtained, which is unfalsified for the purpose of state estimation and optimal control with respect to the available information. However, since the EKF-based parameter estimation method discussed in Section 2.2 is used within this algorithm, a final calibration of the parameters may be needed at this point. More specifically, the EKF-based method

(estimation in a PE setting) tends to emphasize the one-step-ahead prediction capabilities of the model, which means that, because a model with good long-term prediction capabilities is needed for optimal control, e.g. by means of MPC, the parameters should be re-calibrated with an estimation method that emphasizes the pure simulation capabilities of the model (estimation in an OE setting). This should, however, only be done, if it is reasonable to assume that the diffusion term is no longer significant. This is discussed in more detail in the comparison between PE and OE estimation given in Chapter 3. The use of the grey-box modelling algorithm is illustrated in the following example.

Example 2.9 (Developing an unfalsified fermentation process model)
This example illustrates how the grey-box modelling algorithm can be used to develop an unfalsified model from the preliminary fermentation process model shown in Example 2.3. In Examples 2.3-2.8 the first seven steps of the first iteration through the algorithm have already been illustrated, and it has been determined that, to improve its quality, the model should be altered in accordance with the functional relation between μ and S revealed in Figure 2.5b, which is indicative of a growth rate that can be described by Monod kinetics with substrate inhibition. Altering the preliminary model to reflect this in Step 8 gives a model with the system equation:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \ , \ t \in [t_0, t_f] \tag{2.59}$$

where $\mu(S)$ is given by:

$$\mu(S) = \mu_{\max} \frac{S}{K_2 S^2 + S + K_1} \tag{2.60}$$

and the measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{2.61}$$

Returning to Step 3 for the second iteration through the algorithm, and using **CTSM** and the same data set as in Example 2.3, the estimates (and standard deviations and t-scores) shown in Table 2.3 are obtained. To obtain information about the quality of the resulting model, cross-validation residual analysis is performed in Step 4 as shown in Figure 2.6, and the results of this analysis show that both the one-step-ahead prediction capabilities and the pure simulation capabilities of the altered model are very good, which is indicated by the fact that the residuals can all be regarded as white noise. Moving to Step 5, the model is thus unfalsified for the purpose of state estimation and optimal control with respect to the available information, and the model development procedure can be terminated. However, since marginal t-tests for parameter insignificance (see Table 2.3) show that, on a 5% level, there are now no significant parameters in the diffusion term, which is confirmed by a test for simultaneous insignificance based on Wald's W-statistic, the parameters of the model should ideally be re-calibrated at this point with an estimation method that emphasizes the pure simulation capabilities of the model, but this is omitted. ■

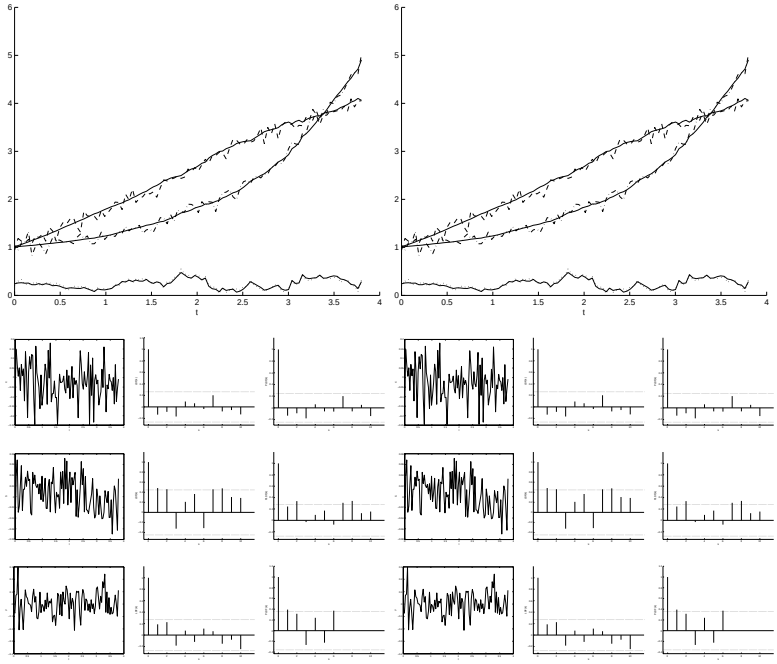


Figure 2.6. Cross-validation residual analysis results for the model in Example 2.9 with parameters in Table 2.3 using the validation data set shown in Figure 2.1b. Top left: One-step-ahead prediction comparison (solid lines: Predicted values); top right: Pure simulation comparison (solid lines: Simulated values); bottom left: One-step-ahead prediction residuals, LDF and PLDF for y_1, y_2 and y_3; bottom right: Pure simulation residuals, LDF and PLDF for y_1, y_2 and y_3.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0148E+00	1.0813E-02	93.8515	Yes
S_0	2.4127E-01	9.4924E-03	25.4177	Yes
V_0	1.0072E+00	8.7723E-03	114.8168	Yes
$\mu_{\max}$	1.0305E+00	1.7254E-02	59.7225	Yes
K_1	3.7929E-02	4.1638E-03	9.1092	Yes
K_2	5.4211E-01	2.4949E-02	21.7286	Yes
σ_{11}	2.3250E-10	2.1044E-07	0.0011	No
σ_{22}	1.4486E-07	7.9348E-05	0.0018	No
σ_{33}	3.2842E-12	3.6604E-09	0.0009	No
S_{11}	7.4828E-03	1.0114E-03	7.3982	Yes
S_{22}	1.0433E-03	1.4331E-04	7.2804	Yes
S_{33}	1.1359E-02	1.6028E-03	7.0867	Yes

Table 2.3. Estimation results. Model in (2.59)-(2.61) - data from Figure 2.1a.

The paper included in Appendix E contains a condensed outline of the material presented in this chapter with a generalized version of the grey-box modelling algorithm presented here. This generalized version is not limited to modelling of fed-batch processes for the purpose of state estimation and optimal control but can be applied to model a variety of systems for different purposes. In this paper a case study extending the examples presented here is also given, and this case study demonstrates that the algorithm can also be successfully applied, when all state variables of a model cannot be measured directly. Additional examples of the application of the algorithm are given in Chapter 3.

2.7.2 Key features and limitations

A key feature of the grey-box modelling algorithm and thus of the proposed grey-box modelling framework as a whole is the possibility of systematically pinpointing and repairing model deficiencies. This is a very powerful feature not shared by other approaches to grey-box modelling reported in literature, e.g. the approach by Bohlin and Graebe (1995) and Bohlin (2001). As mentioned in Section 1.2 the idea of that approach also is to find the simplest model for a given purpose (not necessarily state estimation and optimal control of fed-batch processes), which is consistent with prior physical knowledge and not falsified by available experimental data, and this is done by formulating a sequence of hypothetical model structures of increasing complexity and systematically expanding the model by falsifying incorrect hypotheses through statistical tests based on the experimental data. However, as discussed by Bohlin (2001), a drawback of this approach is that it relies on the model maker to formulate the hypothetical model structures to be tested, which poses the problem that the model maker may run out of ideas for improvement before a sufficiently accurate model is obtained. This problem can be avoided with the framework proposed here due to the feature mentioned above, because it allows the model maker to formulate new hypotheses in an intelligent manner based on information extracted from experimental data. In other words, the proposed framework relies less on the model maker, and, in this particular sense, is more systematic than the approach by Bohlin and Graebe (1995) and Bohlin (2001).

The proposed grey-box modelling framework is, however, not independent of the model maker, and if the model maker is unable to select specific suspect phenomena for further investigation when model deficiencies have been indicated, it is not possible to pinpoint and subsequently repair these deficiencies either. Moreover, like other approaches to grey-box modelling, the performance of the proposed framework is limited by the quality and amount of available prior physical knowledge and experimental data. If there is insufficient prior physical knowledge available to establish an initial model structure, it may not be worthwhile to use this approach as opposed to a data-driven modelling approach, and if the available experimental data is insufficiently informative or if the available measurements render certain subsets of the state variables of the

system unobservable, parameter identifiability may be seriously affected. Because the procedure for pinpointing model deficiencies relies on estimates of the parameters of the diffusion term and because the procedure for subsequently repairing these deficiencies requires that the state variables of the system are observable, the reliability of these procedures may be affected as well. In particular, a situation may occur, where the model is falsified, but where none of the parameters of the diffusion term appear to be significant and pinpointing a specific model deficiency is impossible. A situation may also occur, where the model is falsified and the significance of certain parameters of the diffusion term have allowed a specific deficiency to be pinpointed, but where appropriate estimates of functional relations cannot be obtained to indicate how to repair this deficiency. Both situations imply that a point has been reached, where the model cannot be further improved with the available information. In addition to stressing the need for developing appropriate methods for experimental design to ensure that sufficient information is obtained, which is, however, outside the scope of the work presented in this thesis, this raises a very important question. More specifically, assuming that a "true" model exists, where all state variables are observable, and that the available experimental data is sufficiently informative to ensure that all parameters are identifiable, will the grey-box modelling algorithm then converge to yield the "true" model? In the general case, no rigorous proof of such convergence exists, but the examples presented throughout this chapter have demonstrated that the algorithm may in fact converge for certain simple systems, and the application examples given in Chapter 3 provide additional evidence to support this conclusion.

3

Application examples

In this chapter a number of application examples are given to demonstrate the strengths of the proposed grey-box modelling framework. The first example only focuses on the parameter estimation element of the grey-box modelling cycle, whereas the rest focus on the cycle as a whole and on the related algorithm for systematic iterative model improvement presented in Chapter 2.

3.1 A comparison of PE and OE estimation

As discussed in Chapter 1, the use of continuous-discrete stochastic state space models facilitates the combination of modelling based on prior physical insights with statistical methods for structural identification, parameter estimation and model quality evaluation, which is a key advantage of grey-box modelling. An important aspect in this regard is the fact that continuous-discrete stochastic state space models provide a decomposition of the noise affecting the system into a process noise term (the diffusion term) and a measurement noise term. This facilitates estimation of unknown parameters in a PE setting, which tends to give less biased and more reproducible results than estimation in an OE setting, which is the most commonly used methodology for estimation of parameters in continuous time systems. More specifically, the advantages of PE estimation methods such as the one used within the proposed grey-box modelling framework are due to the fact that process noise can be explicitly accounted for, whereas for OE estimation methods it cannot and is therefore absorbed into the parameter estimates, resulting in significant bias.

To demonstrate the advantages of PE estimation over OE estimation in the presence of process noise and to further discuss the implications, a comparison of the two methods is given here. The PE estimation method used for the comparison is the estimation method used within the proposed grey-box modelling framework and has already been thoroughly discussed in Chapter 2 along with the implementation of this method within the computer program **CTSM**, a detailed account of which is given in Appendix A. The OE estimation method used for the comparison is a standard *nonlinear least squares* (NLS) method applied to an ODE model (Bard, 1974), and this method has been implemented

in MATLAB. Within this method, the system equation is given as follows:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \ , \ t \in [t_0, t_N] \tag{3.1}$$

and the corresponding measurement equation is given as follows:

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{3.2}$$

where $\boldsymbol{y}_k$ is a vector of output variables and $\{\boldsymbol{e}_k\}$ is a white noise process. In other words, the model resembles the continuous-discrete stochastic state space model, except for the fact that the SDE's of the system equation have been replaced with ODE's. Given a sequence of measurements $\boldsymbol{y}_0$, $\boldsymbol{y}_1$, …, $\boldsymbol{y}_k$, …, $\boldsymbol{y}_N$, the objective function for standard NLS can be written as follows:

$$\Phi = \sum_{k=0}^{N} (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0})^T (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0}) \tag{3.3}$$

where $\hat{\boldsymbol{y}}_{k|0}$ is determined by solving the ODE's of the system equation and subsequently applying the measurement equation for a given set of initial conditions $\boldsymbol{x}_0$ and parameter values $\boldsymbol{\theta}$. The parameter estimates are determined by minimizing this function using a nonlinear optimisation algorithm. To avoid numerical approximation, e.g. by means of a set of finite differences, the gradient of the objective function can be computed as follows (Bard, 1974):

$$\begin{aligned} \frac{\partial \Phi}{\partial \boldsymbol{\theta}^T} &= \sum_{k=0}^{N} \frac{\partial ((\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0})^T (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0}))}{\partial \boldsymbol{\theta}^T} \\ &= \sum_{k=0}^{N} \frac{\partial ((\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0})^T (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0}))}{\partial \hat{\boldsymbol{y}}_{k|0}^T} \frac{D\hat{\boldsymbol{y}}_{k|0}}{D\boldsymbol{\theta}^T} \\ &= -2 \sum_{k=0}^{N} (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|0})^T \frac{D\hat{\boldsymbol{y}}_{k|0}}{D\boldsymbol{\theta}^T} \end{aligned} \tag{3.4}$$

where:

$$\begin{aligned} \frac{D\hat{\boldsymbol{y}}_{k|0}}{D\boldsymbol{\theta}^T} &= \frac{D\boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})}{D\boldsymbol{\theta}^T} \\ &= \frac{\partial \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} + \frac{\partial \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})}{\partial \boldsymbol{x}_k^T} \frac{\partial \boldsymbol{x}_k}{\partial \boldsymbol{\theta}^T} \end{aligned} \tag{3.5}$$

and where $\frac{\partial \boldsymbol{x}_k}{\partial \boldsymbol{\theta}^T} = \left(\frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T}\right)_{t=t_k}$ satisfies the following set of ODE's:

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T}\right) &= \frac{\partial}{\partial \boldsymbol{\theta}^T}\left(\frac{d\boldsymbol{x}_t}{dt}\right) = \frac{D\boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{D\boldsymbol{\theta}^T} \\ &= \frac{\partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} + \frac{\partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{\partial \boldsymbol{x}_t^T} \frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T} \ , \ t \in [t_0, t_N] \end{aligned} \tag{3.6}$$

These are the so-called *sensitivity equations*, which can be solved along with the ODE's of the model to yield the gradient of the objective function. Initial conditions for solving these equations can be found as follows (Bard, 1974):

$$\left(\frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T}\right)_{t=t_0} = \frac{\partial \boldsymbol{x}_0}{\partial \boldsymbol{\theta}^T} \tag{3.7}$$

The comparison of this OE estimation method and the PE estimation method of the proposed grey-box modelling framework is given in the following example.

Example 3.1 (A comparison of PE and OE estimation)
This example serves to demonstrate the advantages of PE estimation over OE estimation in the presence of process noise. The estimation problem considered is that of estimating the parameters $\mu_{\max}$ and K_1 (K_2 is fixed at its true value to ensure convergence of the OE estimation method applied) and the initial conditions X_0, S_0 and V_0 in the fermentation process model described in Example 1.1 using the data sets in Figures 2.1-2.3, which have been generated with the continuous-discrete stochastic state space model described in Example 2.1 using different levels of process noise.

For the PE estimation part of the comparison, the estimation method implemented in **CTSM** is applied using a model structure similar to the one described in Example 2.1, where, because additional diffusion term and measurement noise term parameters are also estimated in this case, the complete parameter vector can be written as follows:

$$\boldsymbol{\theta} = \begin{bmatrix} X_0 & S_0 & V_0 & \mu_{\max} & K_1 & \sigma_{11} & \sigma_{22} & \sigma_{33} & S_{11} & S_{22} & S_{33} \end{bmatrix}^T \tag{3.8}$$

For the OE estimation part of the comparison, the standard NLS method described above is applied using a model structure where the system equation is given by:

$$\frac{d}{dt}\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix}, \; t \in [t_0, t_f] \tag{3.9}$$

where the biomass growth rate $\mu(S)$ is given as follows:

$$\mu(S) = \mu_{\max}\frac{S}{K_2 S^2 + S + K_1} \tag{3.10}$$

and where the corresponding measurement equation is given by:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \tag{3.11}$$

where y_1, y_2 and y_3 are output variables and $\{\boldsymbol{e}_k\}$ is a white noise process. The objective function is given by (3.3) and the parameter vector can be written as follows:

$$\boldsymbol{\theta} = \begin{bmatrix} X_0 & S_0 & V_0 & \mu_{\max} & K_1 \end{bmatrix}^T \tag{3.12}$$

The gradient of the objective function, which is given by (3.4), is particularly simple to compute in this specific case, because of the following set of identities:

$$\begin{aligned} \frac{\partial \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} &= \boldsymbol{0} \\ \frac{\partial \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})}{\partial \boldsymbol{x}_k^T} &= \boldsymbol{I} \end{aligned} \tag{3.13}$$

which makes (3.5) identical to the solution to the sensitivity equations, i.e.:

$$\frac{d}{dt}\left(\frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T}\right) = \frac{\partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} + \frac{\partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{\partial \boldsymbol{x}_t^T}\frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T} \ , \ t \in [t_0, t_f] \tag{3.14}$$

where:

$$\begin{aligned} \frac{\partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} &= \begin{bmatrix} 0 & 0 & 0 & \frac{S}{K_2S^2+S+K_1}X & -\frac{\mu_{\max}S}{(K_2S^2+S+K_1)^2}X \\ 0 & 0 & 0 & -\frac{S}{K_2S^2+S+K_1}\frac{X}{Y} & \frac{\mu_{\max}S}{(K_2S^2+S+K_1)^2}\frac{X}{Y} \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ \frac{\partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})}{\partial \boldsymbol{x}_t^T} &= \begin{bmatrix} \mu(S) - \frac{F}{V} & \frac{K_1-K_2S^2}{(K_2S^2+S+K_1)^2}X & \frac{FX}{V^2} \\ -\frac{\mu(S)}{Y} & -\frac{K_1-K_2S^2}{(K_2S^2+S+K_1)^2}\frac{X}{Y} - \frac{F}{V} & -\frac{F(S_F-S)}{V^2} \\ 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{3.15}$$

Initial conditions for solving these equations are given as follows in this case:

$$\left(\frac{\partial \boldsymbol{x}_t}{\partial \boldsymbol{\theta}^T}\right)_{t=t_0} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{3.16}$$

The results of the comparison are shown in Tables 3.1-3.3 in the form of estimates of the parameters and initial states. Uncertainty information in terms of standard deviations of the estimates is not given, because, unlike with the PE estimation method, such information is difficult to obtain with the OE estimation method. As a result, the performance of the two methods can only be compared in terms of bias.

Parameter	True value	PE estimate	OE estimate	PE estimate	OE estimate
X_0	1.0000E+00	1.0095E+00	1.0148E+00	9.8576E-01	9.9595E-01
S_0	2.4490E-01	2.3835E-01	2.4431E-01	2.4760E-01	2.3894E-01
V_0	1.0000E+00	1.0040E+00	1.0092E+00	1.0137E+00	1.0160E+00
$\mu_{\max}$	1.0000E+00	1.0022E+00	9.9852E-01	1.0092E+00	1.0184E+00
K_1	3.0000E-02	3.1629E-02	3.1412E-02	3.2624E-02	3.6663E-02
σ_{11}	0.0000E+00	3.6100E-07	-	8.3976E-06	-
σ_{22}	0.0000E+00	4.7385E-07	-	1.9310E-05	-
σ_{33}	0.0000E+00	7.5881E-14	-	1.1389E-06	-
S_{11}	1.0000E-02	7.5248E-03	-	9.2502E-03	-
S_{22}	1.0000E-03	1.0636E-03	-	8.1408E-04	-
S_{33}	1.0000E-02	1.1388E-02	-	8.3280E-03	-

Table 3.1. Comparison of PE estimation (**CTSM**) and OE estimation (standard NLS) for the data sets in Figure 2.1. Left: Batch no. 1, right: Batch no. 2.

The results in Table 3.1 correspond to the data sets in Figure 2.1, where no process noise is present, and show that in this case the two methods perform equally well in the sense that reasonably unbiased estimates of all parameters and initial states are obtained with both methods. The results in Table 3.2 correspond to the data sets in Figure 2.2, where a moderate level of process noise has been used, and these results show that, although some of the PE estimates seem to be biased as well, the OE estimates are now more biased. Finally, the results in Table 3.3, which correspond to the data sets in Figure 2.3, where a high level of process noise has been used, confirm this tendency and show that the OE estimates are now significantly more biased. ■

The advantages of PE estimation over OE estimation in the presence of process noise imply that, unless it is reasonable to assume that significant process noise is not present, PE estimation should be used, because this gives significantly less biased estimates of the unknown parameters. Moreover, PE estimation provides means to obtain uncertainty information in terms of standard deviations of the estimates and facilitates the use of a number of the powerful statistical tools for model quality evaluation and subsequent model improvement which are integral parts of the proposed grey-box modelling framework. However, as discussed in Chapter 2, PE estimation tends to emphasize the one-step-ahead prediction capabilites of the model, because this method essentially minimizes a sum of squared one-step-ahead prediction errors. OE estimation, on the other hand, minimizes a sum of squared pure simulation errors and therefore tends to emphasize the pure simulation capabilities of the model. Thus, if it is reasonable to assume that significant process noise is not present, and if the model must have good long-term prediction capabilities, which is essential if it is to be used for optimal control of a fed-batch process, e.g. by means of MPC, OE estimation should be used for the final calibration of the parameters of the model. For this purpose, the standard NLS method described above may be used, possibly incorporating a weighting scheme to ensure proper scaling of the individual variables, although this is not as straigtforward as with the PE estimation method implemented in **CTSM**, where this is achieved automatically.

Parameter	True value	PE estimate	OE estimate	PE estimate	OE estimate
X_0	1.0000E+00	1.0647E+00	9.8903E-01	1.0213E+00	1.0050E+00
S_0	2.4490E-01	2.8830E-01	9.7122E-02	2.2395E-01	2.1622E-01
V_0	1.0000E+00	9.8870E-01	8.4471E-01	1.0196E+00	1.0360E+00
$\mu_{\max}$	1.0000E+00	1.0126E+00	9.3045E-01	1.0043E+00	1.0208E+00
K_1	3.0000E-02	3.8748E-02	2.0000E-14	6.4524E-02	6.7207E-02
σ_{11}	1.0000E-01	1.0828E-01	-	1.5974E-06	-
σ_{22}	1.0000E-01	1.2294E-01	-	8.2424E-02	-
σ_{33}	1.0000E-01	7.7399E-02	-	9.8385E-02	-
S_{11}	1.0000E-02	8.4982E-03	-	8.9795E-03	-
S_{22}	1.0000E-03	9.3489E-04	-	1.0258E-03	-
S_{33}	1.0000E-02	9.5192E-03	-	8.6510E-03	-

Table 3.2. Comparison of PE estimation (**CTSM**) and OE estimation (standard NLS) for the data sets in Figure 2.2. Left: Batch no. 1, right: Batch no. 2.

3.2 A case with a complex deficiency

The performance of the proposed grey-box modelling framework has already been demonstrated by means of the examples given in Chapter 2, which illustrate the individual elements of the grey-box modelling cycle as well as the corresponding algorithm for systematic iterative model improvement for a simple example. To further demonstrate the performance of the proposed framework, a somewhat more complicated example is considered in the following.

Example 3.2 (A case with a complex deficiency)
This example demonstrates the performance of the proposed grey-box modelling framework for a fed-batch fermentation process represented by a simulation model that describes growth of biomass on two different substrates with multiple Monod kinetics and inhibition by one of the substrates. The model is given as follows:

$$\frac{dX}{dt} = \mu(S_1, S_2)X - \frac{FX}{V} \tag{3.17}$$

$$\frac{dS_1}{dt} = -Y_1\mu(S_1, S_2)X + \frac{F\,(S_{F,1} - S_1)}{V} \tag{3.18}$$

$$\frac{dS_2}{dt} = -Y_2\mu(S_1, S_2)X + \frac{F\,(S_{F,2} - S_2)}{V} \tag{3.19}$$

$$\frac{dV}{dt} = F \tag{3.20}$$

for $t \in [t_0, t_f]$, where X ($\frac{g}{l}$) is the biomass concentration, S_1 ($\frac{g}{l}$) and S_2 ($\frac{g}{l}$) are concentrations of the two substrates, V (l) is the reactor volume, F ($\frac{l}{h}$) is the feed flow rate, $Y_1 = 2$ and $Y_2 = 0.1$ are yield coefficients and $S_{F,1} = 10\frac{g}{l}$ and $S_{F,2}$ ($\frac{g}{l}$) are feed concentrations of the two substrates. $t_0 = 0h$ and $t_f = 3.8h$ are initial and final times of a typical fed-batch run and $\mu(S_1, S_2)$ (h^{-1}) is the biomass growth rate, i.e.:

$$\mu(S_1, S_2) = \mu_{\max}\frac{S_1}{K_{12}S_1^2 + S_1 + K_{11}}\frac{S_2}{S_2 + K_2} \tag{3.21}$$

where $\mu_{\max} = 1h^{-1}$, $K_{11} = 0.03\frac{g}{l}$, $K_{12} = 0.5\frac{l}{g}$ and $K_2 = 0.06\frac{g}{l}$ are kinetic parameters. In order to generate data from this model by perturbing the feed flow rate along

Parameter	True value	PE estimate	OE estimate	PE estimate	OE estimate
X_0	1.0000E+00	9.5255E-01	8.4096E-01	1.0808E+00	1.3441E+00
S_0	2.4490E-01	2.3878E-01	4.5647E-02	2.0078E-01	9.0551E-01
V_0	1.0000E+00	9.8120E-01	1.2504E+00	1.1813E+00	1.6106E+00
$\mu_{\max}$	1.0000E+00	9.6795E-01	8.8212E-01	1.0341E+00	7.9587E-01
K_1	3.0000E-02	3.1606E-02	1.9189E-02	4.4851E-02	6.2200E-12
σ_{11}	3.1623E-01	3.1715E-01	-	2.7136E-01	-
σ_{22}	3.1623E-01	2.7524E-01	-	3.8652E-01	-
σ_{33}	3.1623E-01	2.5364E-01	-	3.9257E-01	-
S_{11}	1.0000E-02	7.9042E-03	-	1.0219E-02	-
S_{22}	1.0000E-03	1.2357E-03	-	1.5330E-04	-
S_{33}	1.0000E-02	8.4691E-03	-	9.7136E-03	-

Table 3.3. Comparison of PE estimation (**CTSM**) and OE estimation (standard NLS) for the data sets in Figure 2.3. Left: Batch no. 1, right: Batch no. 2.

an appropriate trajectory, an optimal such trajectory is first determined by solving a specific productivity maximization problem, which can be stated as follows:

$$\max_{\substack{X_0,S_{10},S_{20},V_0,\\ F(t)\ ,\ t\in[t_0,t_f]}} V(t_f)X(t_f) \tag{3.22}$$

subject to the above model equations. In other words, the problem is to determine the initial conditions and the open loop feed flow rate trajectory that gives optimal productivity in terms of the amount of biomass at the end of a run. By applying an appropriate variable transformation and subsequently using Pontryagin's maximum principle, the following conditions for optimal operation can be obtained:

$$\begin{aligned} 0 &= \frac{\partial\mu(S_1,S_2)}{\partial S_1} = \mu_{\max}\frac{K_{11}-K_{12}S_1^2}{\left(K_{12}S_1^2+S_1+K_{11}\right)^2}\frac{S_2}{S_2+K_2} \Rightarrow S_1 = \sqrt{\frac{K1}{K2}} = S_1^* \\ 0 &= \frac{\partial\mu(S_1,S_2)}{\partial S_2} = \mu_{\max}\frac{S_1}{K_{12}S_1^2+S_1+K_{11}}\frac{K_2}{\left(S_2+K_2\right)^2} \Rightarrow S_2 \to \infty \end{aligned} \tag{3.23}$$

The latter condition is not practically realizable, so $\mu(S_1,S_2)$ can only be maximized with respect to S_1. Assuming that the initial concentration $S_{10} = S_1^*$ and by choosing the feed flow rate in a way that makes $\frac{dS_1}{dt} = 0$, S_1 can be kept at $S_{10} = S_1^*$, i.e.:

$$0 = \frac{dS_1}{dt} = -Y_1\mu(S_{10},S_{20})X + \frac{F\left(S_{F,1}-S_{10}\right)}{V} \Rightarrow F = \frac{Y_1\mu(S_{10},S_{20})XV}{\left(S_{F,1}-S_{10}\right)} \tag{3.24}$$

This expression is inserted into two of the other equations of the original model, i.e.:

$$\begin{aligned} \frac{dX}{dt} &= \mu(S_{10},S_{20})X - \frac{Y_1\mu(S_{10},S_{20})XV}{\left(S_{F,1}-S_{10}\right)}\frac{X}{V} \\ \frac{dV}{dt} &= \frac{Y_1\mu(S_{10},S_{20})XV}{\left(S_{F,1}-S_{10}\right)} \end{aligned} \quad , \begin{matrix} X(t_0) = X_0 \\ V(t_0) = V_0 \end{matrix} \ , \ t \in [t_0,t_f] \tag{3.25}$$

and by setting $a = \mu(S_{10},S_{20})$ and $b = \frac{Y_1\mu(S_{10},S_{20})}{\left(S_{F,1}-S_{10}\right)}$, the equation for X can be solved:

$$\begin{aligned} \frac{dX}{dt} &= aX - bX^2 \\ X &= \frac{ae^{at}c}{1+be^{at}c} \ , \ t \in [t_0,t_f] \end{aligned} \tag{3.26}$$

with $c = \frac{X_0}{a-bX_0}$, whereupon the equation for V can be solved as follows:

$$\begin{aligned} \frac{dV}{dt} &= bXV = b\frac{ae^{at}c}{1+be^{at}c}V \\ V &= \frac{1+be^{at}c}{1+bc}V_0 \ , \ t \in [t_0,t_f] \end{aligned} \tag{3.27}$$

By substituting these solutions back into the equation for the feed flow rate, an analytical expression for the optimal feed flow rate trajectory can be obtained, i.e.:

$$\begin{aligned} F = bXV &= b\frac{ae^{at}c}{1+be^{at}c}\frac{1+be^{at}c}{1+bc}V_0 \\ &= be^{at}X_0V_0 \ , \ t \in [t_0,t_f] \end{aligned} \tag{3.28}$$

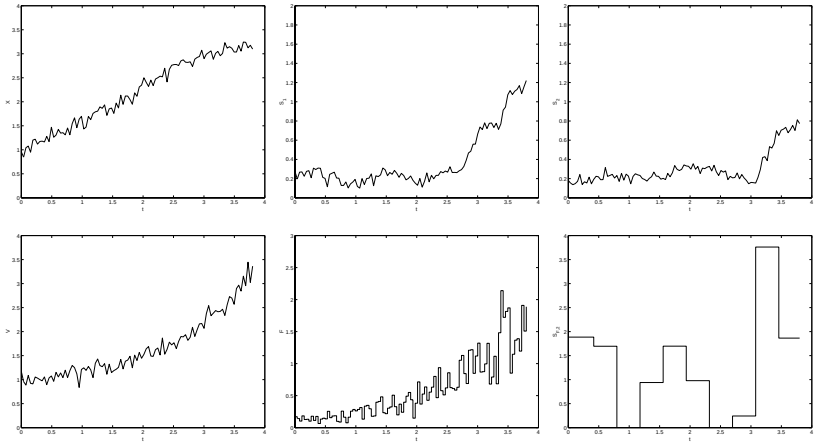


Figure 3.1. Data set no. 1 for Example 3.2. Top: X, S_1, S_2. Bottom: V, F, $S_{F,2}$.

Using perturbed versions of this feed flow rate trajectory (along with low frequency perturbation in $S_{F,2}$), two data sets (shown in Figures 3.1-3.2) are generated by means of stochastic simulation using the Euler scheme (see Example 2.2). For this purpose a re-formulated version of the model is applied, which has the following system equation:

$$d\begin{pmatrix} X \\ S_1 \\ S_2 \\ V \end{pmatrix} = \begin{pmatrix} \mu(S_1,S_2)X - \frac{FX}{V} \\ -Y_1\mu(S_1,S_2)X + \frac{F(S_{F,1}-S_1)}{V} \\ -Y_2\mu(S_1,S_2)X + \frac{F(S_{F,2}-S_2)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.29)$$

where $t \in [t_0, t_f]$, and the following measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}_k = \begin{pmatrix} X \\ S_1 \\ S_2 \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 & 0 \\ 0 & S_{22} & 0 & 0 \\ 0 & 0 & S_{33} & 0 \\ 0 & 0 & 0 & S_{44} \end{bmatrix} \quad (3.30)$$

The specific initial state values applied are $(X_0, S_{10}, S_{20}, V_0) = (1, S_1^*, \frac{1}{2}S_1^*, 1)$, and the parameter values applied are the deterministic parameter values mentioned above, the diffusion term parameter values $\sigma_{11} = \sigma_{22} = \sigma_{33} = \sigma_{44} = 0$ and the measurement noise term parameter values $S_{11} = 0.01$, $S_{22} = 0.001$, $S_{33} = 0.001$ and $S_{44} = 0.01$. A discretization time interval corresponding to $\frac{1}{10000}$ of t_f is used and every 100'th value is sampled (see Example 2.2) to give data sets containing 101 samples each.

Using the generated data sets, the performance of the grey-box modelling cycle and the corresponding algorithm for systematic iterative model improvement is now illustrated by assuming that an initial model structure corresponding to (3.29)-(3.30) is available, where the true structure of the biomass growth rate $\mu(S_1, S_2)$ is unknown.

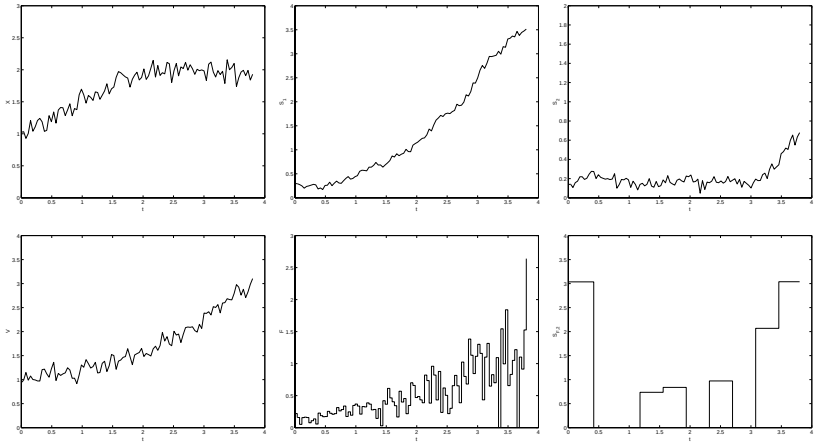


Figure 3.2. Data set no. 2 for Example 3.2. Top: X, S_1, S_2. Bottom: V, F, $S_{F,2}$.

This is a reasonable assumption, because a model of this type can easily be formulated by applying mass balances to derive an ODE model and by translating this model into a continuous-discrete stochastic state space model with a diagonal parameterization of the diffusion term, which is also straightforward. Steps 1 and 2 of the algorithm have thus been completed to yield a model with the following system equation:

$$d\begin{pmatrix} X \\ S_1 \\ S_2 \\ V \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -Y_1\mu X + \frac{F(S_{F,1}-S_1)}{V} \\ -Y_2\mu X + \frac{F(S_{F,2}-S_2)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \tag{3.31}$$

where $t \in [t_0, t_f]$, and where, because the true structure of the biomass growth rate is unknown, a constant growth rate μ has been assumed. The measurement equation of the model is equivalent to (3.30). In Step 3 of the algorithm, the unknown parameters of the model are estimated using **CTSM** and the data set in Figure 3.1, which gives the results shown in Table 3.4. To evaluate the quality of the resulting model in terms of its prediction capabilities, cross-validation residual analysis is performed in Step 4, and, since the intended purpose of the model is assumed to be application for subsequent state estimation and optimal control, which requires a model with good long-term prediction capabilities, only pure simulation residual analysis is performed, cf. Figure 3.3. The results of this analysis show that the model has poor pure simulation capabilities and thus falsify the model for the purpose of optimal control in Step 5, which means that the model development procedure implied by the grey-box modelling cycle must be repeated by re-formulating the model. Step 6 of the algorithm, which deals with pinpointing of model deficiencies, is therefore applied. Table 3.4 includes t-scores for performing marginal tests for insignificance of the individual parameters. On a 5% level, these show that only σ_{44} is insignificant.

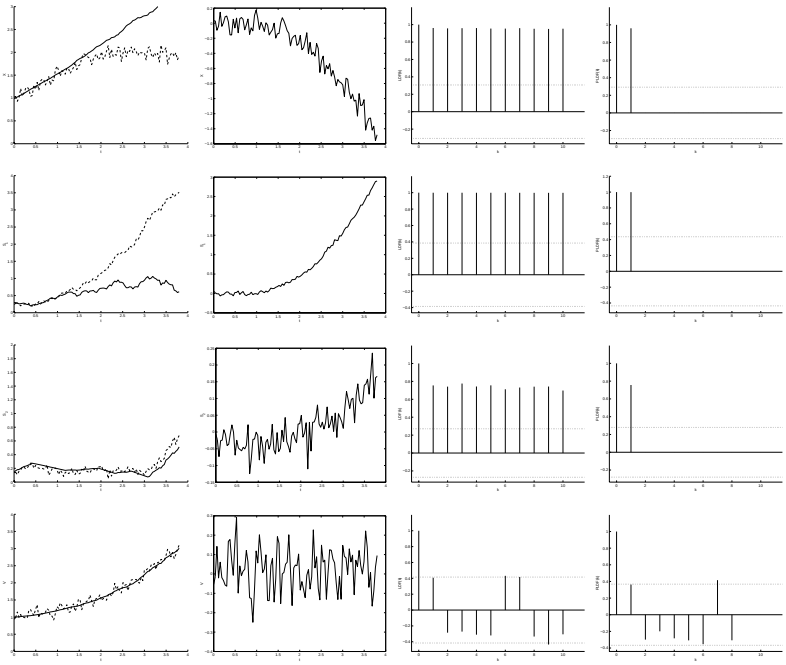


Figure 3.3. Pure simulation cross-validation residual analysis results for the model in (3.31) and (3.30) with parameters in Table 3.4 using the validation data set shown in Figure 3.2. Top-down: y_1, y_2, y_3 and y_4. Left-right: Pure simulation comparison (solid lines: Simulated values), residuals, LDF and PLDF.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.8928E-01	4.0081E-02	24.6819	Yes
S_{10}	2.4057E-01	8.3171E-02	2.8925	Yes
S_{20}	1.4383E-01	3.6991E-02	3.8882	Yes
V_0	9.9274E-01	1.0085E-02	98.4370	Yes
$\mu_{\max}$	6.1743E-01	7.6554E-03	80.6534	Yes
σ_{11}	4.3756E-02	2.1532E-02	2.0321	Yes
σ_{22}	8.1328E-02	1.4821E-02	5.4872	Yes
σ_{33}	3.7169E-02	1.7445E-02	2.1306	Yes
σ_{44}	1.5274E-06	1.8520E-05	0.0825	No
S_{11}	7.8047E-03	1.2265E-03	6.3632	Yes
S_{22}	9.5065E-04	1.7527E-04	5.4239	Yes
S_{33}	1.1190E-03	2.0934E-04	5.3457	Yes
S_{44}	1.1593E-02	1.6556E-03	7.0025	Yes

Table 3.4. Estimation results. Model in (3.31) and (3.30) - data from Figure 3.1.

The fact that the remaining parameters of the diffusion term are all significant, indicates that the corresponding elements of the drift term may be incorrect. These elements all depend on μ, which means that μ is an obvious model deficiency suspect, so to investigate this further, the model is re-formulated with μ as an additional state variable, which yields a model with the following system equation:

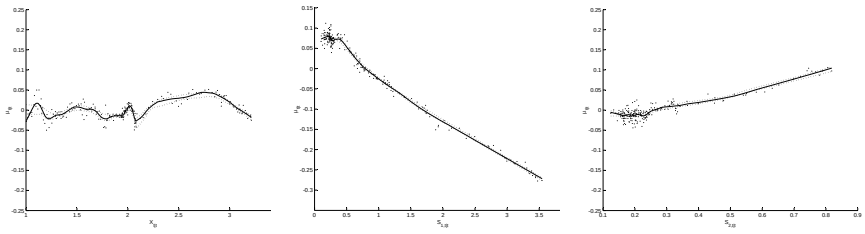
$$d\begin{pmatrix} X \\ S_1 \\ S_2 \\ V \\ \mu \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -Y_1\mu X + \frac{F(S_{F,1}-S_1)}{V} \\ -Y_2\mu X + \frac{F(S_{F,2}-S_2)}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 & 0 \\ 0 & 0 & 0 & \sigma_{44} & 0 \\ 0 & 0 & 0 & 0 & \sigma_{55} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.32)$$

where $t \in [t_0, t_f]$, and where the last element of the drift term is zero, because μ has been assumed to be constant. The measurement equation remains equivalent to (3.30). Estimating the unknown parameters of this model using **CTSM** and the same data set as before, gives the results shown in Table 3.5, and inspection of the t-scores for marginal tests for insignificance now show that, of the parameters of the diffusion term, only σ_{55} is significant. This indicates that there is substantial variation in μ and thus confirms the suspicion that μ is deficient. Moving to Step 7 of the algorithm, nonparametric modelling can now be applied to determine how to improve the model.

Using the re-formulated model in (3.32) and (3.30) and the parameter estimates in Table 3.5, state estimates $\hat{X}_{k|k}$, $\hat{S}_{1,k|k}$, $\hat{S}_{2,k|k}$, $\hat{V}_{k|k}$, $\hat{\mu}_{k|k}$, $k = 0, \ldots, N$, are computed with **CTSM** from the data sets shown in Figures 3.1-3.2 and an additive model is fitted to reveal the true structure of the function describing μ by means of estimates of functional relations between μ and the original state variables. It is reasonable to make the assumption that μ does not depend on V, so only functional relations between $\hat{\mu}_{k|k}$ and $\hat{X}_{k|k}$, $\hat{S}_{1,k|k}$ and $\hat{S}_{2,k|k}$ are estimated, which gives the results shown in Figure 3.4. These plots indicate that $\hat{\mu}_{k|k}$ does not depend on $\hat{X}_{k|k}$, but is highly

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0043E+00	1.2949E-02	77.5607	Yes
S_{10}	2.4473E-01	1.2938E-02	18.9150	Yes
S_{20}	1.2464E-01	5.1975E-03	23.9802	Yes
V_0	9.9527E-01	8.5839E-03	115.9467	Yes
μ_0	5.9384E-01	3.9559E-02	15.0115	Yes
σ_{11}	2.2203E-06	9.1593E-06	0.2424	No
σ_{22}	1.8052E-06	7.3434E-06	0.2458	No
σ_{33}	2.4187E-07	1.0447E-06	0.2315	No
σ_{44}	5.8310E-11	3.6366E-10	0.1603	No
σ_{55}	5.3179E-02	1.4390E-02	3.6955	Yes
S_{11}	7.4298E-03	1.0513E-03	7.0673	Yes
S_{22}	1.1182E-03	1.7492E-04	6.3928	Yes
S_{33}	1.3616E-03	1.8904E-04	7.2027	Yes
S_{44}	1.1529E-02	1.5798E-03	7.2978	Yes

Table 3.5. Estimation results. Model in (3.32) and (3.30) - data from Figure 3.1.



(a) $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$. (b) $\hat{\mu}_{k|k}$ vs. $\hat{S}_{1,k|k}$. (c) $\hat{\mu}_{k|k}$ vs. $\hat{S}_{2,k|k}$.

Figure 3.4. Partial dependence plots of $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$, $\hat{S}_{1,k|k}$ and $\hat{S}_{2,k|k}$ obtained by applying additive model fitting using locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths determined using 5-fold cross-validation). Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals computed from 1000 replicates (see Appendix C for details).

dependent on $\hat{S}_{1,k|k}$ and slightly less dependent on $\hat{S}_{2,k|k}$. Because of the apparent dependence on more than one variable, further investigations are needed to rule out the possibility that this is caused by an actual dependence on e.g. the product of these variables or a fraction between them, but performing such investigations shows that this does not seem to be the case here. Instead, since the apparent dependence on more than one variable may be due to other types of correlations as well, only the strongest dependence, i.e. the dependence on $\hat{S}_{1,k|k}$, is taken into account. In Step 8 of the algorithm, the model is therefore re-formulated by replacing the assumption of constant μ with an assumption of μ being a function of S_1 that complies with the functional relation revealed in Figure 3.4b. This relation is indicative of a biomass growth rate that is governed by Monod kinetics and strongly inhibited by the first substrate, which makes it reasonable to assume the following functional form:

$$\mu(S_1) = \mu_{\max} \frac{S_1}{K_{12} S_1^2 + S_1 + K_{11}} \tag{3.33}$$

and hence the following system equation:

$$d\begin{pmatrix} X \\ S_1 \\ S_2 \\ V \end{pmatrix} = \begin{pmatrix} \mu(S_1)X - \frac{FX}{V} \\ -Y_1\mu(S_1)X + \frac{F(S_{F,1}-S_1)}{V} \\ -Y_2\mu(S_1)X + \frac{F(S_{F,2}-S_2)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \tag{3.34}$$

where $t \in [t_0, t_f]$. The measurement equation remains equivalent to (3.30). Returning to Step 3 of the algorithm, the unknown parameters of the new model are estimated using **CTSM** and the data set in Figure 3.1, which gives the results shown in Table 3.6, and in Step 4 the quality of the resulting model is evaluated by performing cross-validation residual analysis, cf. Figure 3.5. The results of this analysis show that the new model has poor pure simulation capabilities as well, and in Step 5 of the algorithm this model is therefore also falsified for the purpose of optimal control.

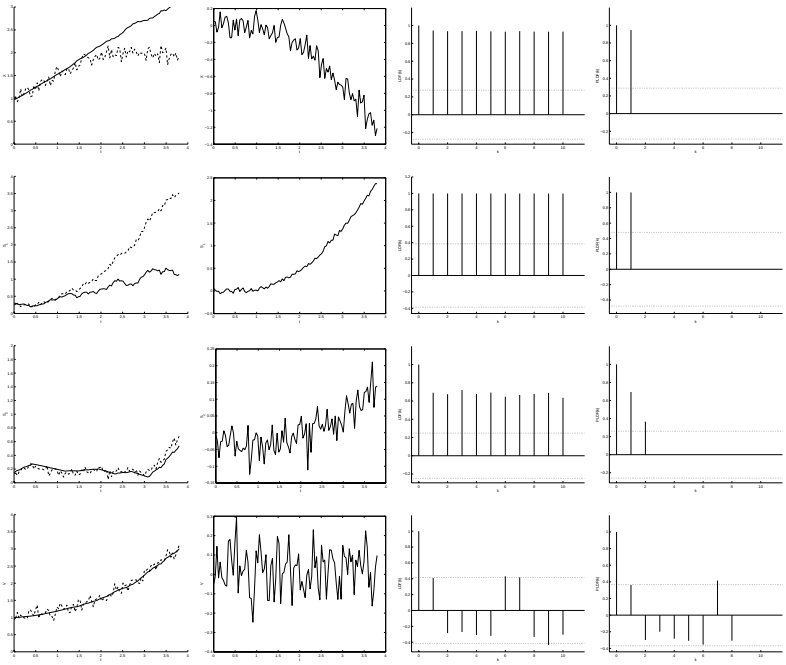


Figure 3.5. Pure simulation cross-validation residual analysis results for the model in (3.34) and (3.30) with parameters in Table 3.6 using the validation data set shown in Figure 3.2. Top-down: y_1, y_2, y_3 and y_4. Left-right: Pure simulation comparison (solid lines: Simulated values), residuals, LDF and PLDF.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.7252E-01	1.5610E-02	62.3021	Yes
S_{10}	2.4155E-01	7.0201E-02	3.4409	Yes
S_{20}	1.4480E-01	4.2272E-02	3.4254	Yes
V_0	9.9031E-01	1.1358E-02	87.1905	Yes
$\mu_{\max}$	6.8920E-01	1.6226E-01	4.2476	Yes
K_{11}	8.7882E-03	4.2577E-02	0.2064	No
K_{12}	1.8640E-01	2.8336E-01	0.6578	No
σ_{11}	2.4387E-07	1.2018E-05	0.0203	No
σ_{22}	6.1827E-02	1.9015E-02	3.2514	Yes
σ_{33}	4.0159E-02	1.7820E-02	2.2536	Yes
σ_{44}	1.7596E-09	8.0415E-08	0.0219	No
S_{11}	7.8187E-03	1.1953E-03	6.5411	Yes
S_{22}	1.0090E-03	1.8316E-04	5.5091	Yes
S_{33}	1.0998E-03	2.0803E-04	5.2868	Yes
S_{44}	1.1499E-02	1.6922E-03	6.7953	Yes

Table 3.6. Estimation results. Model in (3.34) and (3.30) - data from Figure 3.1.

In other words, the new model does not seem to provide significant improvement in terms of prediction capabilities in comparison with the original model. Before Step 6 of the algorithm, which deals with pinpointing of model deficiencies, is applied, statistical tests are therefore performed to investigate if the replacement of the assumption of a constant biomass growth rate μ with the assumption of $\mu(S_1)$ in (3.33) has in fact been insignificant. Table 3.6 includes t-scores for performing marginal tests for insignificance of the individual parameters, which show that, on a 5% level, neither K_{11} nor K_{12} is significant. If this is indeed the case, meaning that these parameters may be eliminated by setting them equal to zero, (3.33) reduces to $\mu(S_1) = \mu_{\max}$, which is equivalent to an assumption of constant μ, and hence proves that the new model is not significantly different from the original. However, because these two marginal tests do not take correlations into account, such inference cannot be made. Instead a test based on Wald's W-statistic is performed to test the hypothesis:

$$H_0: \quad \begin{pmatrix} K_{11} \\ K_{12} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3.35}$$

against the corresponding alternative:

$$H_1: \quad \begin{pmatrix} K_{11} \\ K_{12} \end{pmatrix} \neq \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3.36}$$

i.e. to test whether the two parameters are simultaneously insignificant or not. The test quantity can be computed from the t-scores for the two parameters and the relevant part of the corresponding correlation matrix as follows (see Appendix B):

$$W(\hat{K}_{11}, \hat{K}_{12}) = \begin{bmatrix} 0.2064 & 0.6578 \end{bmatrix} \begin{bmatrix} 1 & 0.9930 \\ 0.9930 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 0.2064 \\ 0.6578 \end{bmatrix} = 14.74 \tag{3.37}$$

The critical area for a test on a 5% level is $W(\hat{K}_{11}, \hat{K}_{12}) > \chi^2(2)_{0.95} = 5.991$. In other words, the H_0 hypothesis is rejected, which means that, simultaneously, the two parameters are significant. This proves that the new model is in fact significantly different from the original and indicates that the S_1-dependent part of the expression for the biomass growth rate should be retained. Moving on with Step 6 of the algorithm, the t-scores included in Table 3.6 show that two of the parameters of the diffusion term are significant, i.e. σ_{22} and σ_{33}, and this indicates that the corresponding elements of the drift term may be incorrect. These elements both depend on $\mu(S_1)$, which is thus a candidate for being deficient. To investigate this further, the model should therefore be re-formulated with $\mu(S_1)$ as an additional state variable. However, prior analysis (see Figure 3.4) has shown potential dependence of the biomass growth rate on both S_1 and S_2 and the above analysis has indicated that the already modelled S_1-dependence should be retained. Therefore, only $\mu_{\max}$ is included as an additional state variable to yield a model with the following system equation:

$$d\begin{pmatrix} X \\ S_1 \\ S_2 \\ V \\ \mu_{\max} \end{pmatrix} = \begin{pmatrix} \mu(S_1)X - \frac{FX}{V} \\ -Y_1\mu(S_1)X + \frac{F(S_{F,1}-S_1)}{V} \\ -Y_2\mu(S_1)X + \frac{F(S_{F,2}-S_2)}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 & 0 \\ 0 & 0 & 0 & \sigma_{44} & 0 \\ 0 & 0 & 0 & 0 & \sigma_{55} \end{bmatrix} d\boldsymbol{\omega}_t \tag{3.38}$$

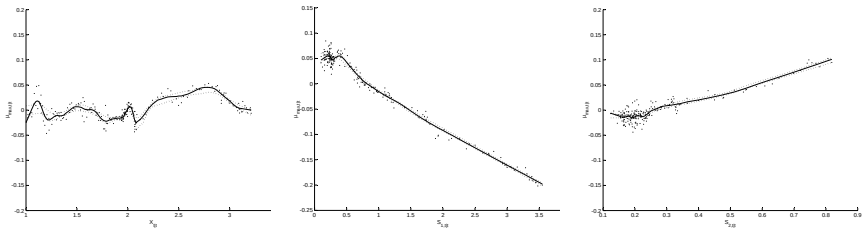
where $t \in [t_0, t_f]$, and where the last element of the drift is zero, because $\mu_{\max}$ has been assumed to be constant. The measurement equation remains equivalent to (3.30). Estimating the unknown parameters of this model using **CTSM** and the same data set as before, gives the results shown in Table 3.7, and inspection of the t-scores for marginal tests for insignificance now show that, of the parameters of the diffusion term, only σ_{55} is significant. This indicates that there is substantial variation in $\mu_{\max}$ and thus confirms the suspicion that $\mu_{\max}$ is deficient. Moving to Step 7 of the algorithm, nonparametric modelling can now be applied to improve the model.

Using the re-formulated model in (3.38) and (3.30) and the parameter estimates in Table 3.7, state estimates $\hat{X}_{k|k}$, $\hat{S}_{1,k|k}$, $\hat{S}_{2,k|k}$, $\hat{V}_{k|k}$, $\hat{\mu}_{\max,k|k}$, $k = 0, \ldots, N$, are computed with **CTSM** from the data sets shown in Figures 3.1-3.2 and an additive model is fitted to reveal the true structure of the function describing $\mu_{\max}$ by means of estimates of functional relations between $\mu_{\max}$ and the original state variables. It is reasonable to assume that $\mu_{\max}$ does not depend on V, so only functional relations between $\hat{\mu}_{\max,k|k}$ and $\hat{X}_{k|k}$, $\hat{S}_{1,k|k}$ and $\hat{S}_{2,k|k}$ are estimated, which gives the results shown in Figure 3.6. These plots resemble the plots in Figure 3.4 by indicating that $\hat{\mu}_{\max,k|k}$ is independent of $\hat{X}_{k|k}$ but highly dependent on $\hat{S}_{1,k|k}$ and slightly less dependent on $\hat{S}_{2,k|k}$, and further investigations indicate that the apparent dependence on more than one variable does not seem to be caused by an actual dependence on e.g. the product of these variables or a fraction between them. More likely, this dependence is due to the fact that some of the variations in the already modelled S_1-dependent part of the expression for the biomass growth rate are absorbed into $\mu_{\max}$ (note that the estimates of K_{11} and K_{12} have changed from Table 3.6 to Table 3.7).

Thus assuming that the dependence on $\hat{S}_{1,k|k}$ has already been adequately accounted for, only the dependence on $\hat{S}_{2,k|k}$ is therefore taken into account. In Step 8 of the algorithm, the model is therefore re-formulated by replacing the assumption of

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0039E+00	2.0273E-02	49.5186	Yes
S_{10}	2.4453E-01	1.4719E-02	16.6136	Yes
S_{20}	1.2458E-01	7.1382E-03	17.4524	Yes
V_0	9.9489E-01	1.9002E-02	52.3575	Yes
$\mu_{\max,0}$	6.1176E-01	6.6621E-02	9.1828	Yes
K_{11}	3.0850E-14	4.3363E-11	0.0007	No
K_{12}	1.0826E-01	9.4352E-02	1.1475	No
σ_{11}	9.9716E-07	4.2966E-04	0.0023	No
σ_{22}	1.4180E-06	6.9594E-04	0.0020	No
σ_{33}	1.2599E-05	4.9623E-03	0.0025	No
σ_{44}	2.5428E-14	2.8508E-11	0.0009	No
σ_{55}	4.8391E-02	1.3997E-02	3.4573	Yes
S_{11}	7.4332E-03	1.2088E-03	6.1493	Yes
S_{22}	1.1189E-03	3.1452E-04	3.5574	Yes
S_{33}	1.3631E-03	2.5160E-04	5.4178	Yes
S_{44}	1.1514E-02	1.4838E-03	7.7602	Yes

Table 3.7. Estimation results. Model in (3.38) and (3.30) - data from Figure 3.1.



(a) $\hat{\mu}_{\max,k|k}$ vs. $\hat{X}_{k|k}$. (b) $\hat{\mu}_{\max,k|k}$ vs. $\hat{S}_{1,k|k}$. (c) $\hat{\mu}_{\max,k|k}$ vs. $\hat{S}_{2,k|k}$.

Figure 3.6. Partial dependence plots of $\hat{\mu}_{\max,k|k}$ vs. $\hat{X}_{k|k}$, $\hat{S}_{1,k|k}$ and $\hat{S}_{2,k|k}$ obtained by applying additive model fitting using locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths determined using 5-fold cross-validation). Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals computed from 1000 replicates (see Appendix C for details).

constant $\mu_{\max}$ with an assumption of $\mu_{\max}$ being a function of S_2 that complies with the functional relation revealed in Figure 3.6c. The increasing tendency in this plot is indicative of a functionality that can be described by an expression of the Monod type (this may be percieved as conjecture but is supported by the fact that bioprocesses are often governed by kinetics of this type), which makes it reasonable to assume the following functional form for the complete expression for the biomass growth rate:

$$\mu(S_1, S_2) = \mu_{\max} \frac{S_1}{K_{12} S_1^2 + S_1 + K_{11}} \frac{S_2}{S_2 + K_2} \tag{3.39}$$

and hence the following system equation:

$$d \begin{pmatrix} X \\ S_1 \\ S_2 \\ V \end{pmatrix} = \begin{pmatrix} \mu(S_1, S_2) X - \frac{FX}{V} \\ -Y_1 \mu(S_1, S_2) X + \frac{F(S_{F,1} - S_1)}{V} \\ -Y_2 \mu(S_1, S_2) X + \frac{F(S_{F,2} - S_2)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \tag{3.40}$$

where $t \in [t_0, t_f]$. The measurement equation remains equivalent to (3.30). Returning to Step 3 of the algorithm, the unknown parameters of the new model are estimated using **CTSM** and the data set in Figure 3.1, which gives the results shown in Table 3.8, and in Step 4 the quality of the resulting model is evaluated by performing cross-validation residual analysis, cf. Figure 3.7. The results of this analysis show that the model has significantly better pure simulation capabilities than the previously analyzed models. More specifically, the y_1, y_3 and y_4 residuals can be regarded as white noise, and the y_2 pure simulation comparison is much better than with the previously analyzed models. However, there seems to be some non-random variation still left in the y_2 residuals. Depending on the specific degree of accuracy required, which is essentially an application-specific and therefore often subjective measure, the model may thus be falsified for the purpose of optimal control in Step 5, meaning that

the model development procedure must be repeated by re-formulating the model, but this is assumed not to be the case. Furthermore, all information available in the data set used for estimation has been exhausted in the context of the proposed grey-box modelling framework, because a model has been developed where the diffusion term is insignificant[1], which means that model deficiencies can no longer be systematically pinpointed. Moreover, the true model in (3.29)-(3.30) has been recovered. ■

The above example demonstrates the performance of the proposed grey-box modelling framework for a model with a more complex deficiency than the one used in the examples given in Chapter 2. In particular, the example demonstrates that a deficiency caused by an incorrectly modelled function of more than one variable can also be repaired by applying the methods of the proposed grey-box modelling cycle and the corresponding algorithm for systematic iterative model improvement. However, the example also demonstrates that model development may be much more complicated in such cases due to correlation effects, which may lead to misinterpretation of results in the sense that, unless proper precautions are taken, variations in some variables may be incorrectly interpreted as variations in other variables, which may limit the performance of the proposed framework by increasing the number of iterations through the modelling cycle needed to develop a model with sufficient accuracy.

[1]Inspection of the t-scores for marginal tests for insignificance (Table 3.8) suggest that, on a 5% level, there are no significant parameters in the diffusion term, which is confirmed by a test for simultaneous insignificance based on Wald's W-statistic. A final calibration of the remaining model parameters should therefore ideally be performed at this stage, using an estimation method that emphasizes the pure simulation capabilities of the model.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0093E+00	1.1575E-02	87.1990	Yes
S_{10}	2.3284E-01	9.3650E-03	24.8631	Yes
S_{20}	1.2352E-01	5.4266E-03	22.7616	Yes
V_0	9.9461E-01	8.8033E-03	112.9807	Yes
$\mu_{\max}$	1.0421E+00	6.5420E-02	15.9301	Yes
K_{11}	3.8553E-02	1.0952E-02	3.5200	Yes
K_{12}	5.5257E-01	8.8254E-02	6.2611	Yes
K_2	6.3228E-02	7.5480E-03	8.3768	Yes
σ_{11}	1.7046E-06	1.8305E-05	0.0931	No
σ_{22}	7.1101E-10	1.4125E-08	0.0503	No
σ_{33}	1.9722E-10	4.7941E-09	0.0411	No
σ_{44}	5.2778E-10	1.0034E-08	0.0526	No
S_{11}	7.4408E-03	1.0405E-03	7.1511	Yes
S_{22}	1.0342E-03	1.5105E-04	6.8471	Yes
S_{33}	1.3603E-03	2.0785E-04	6.5443	Yes
S_{44}	1.1519E-02	1.5025E-03	7.6665	Yes

Table 3.8. Estimation results. Model in (3.40) and (3.30) - data from Figure 3.1.

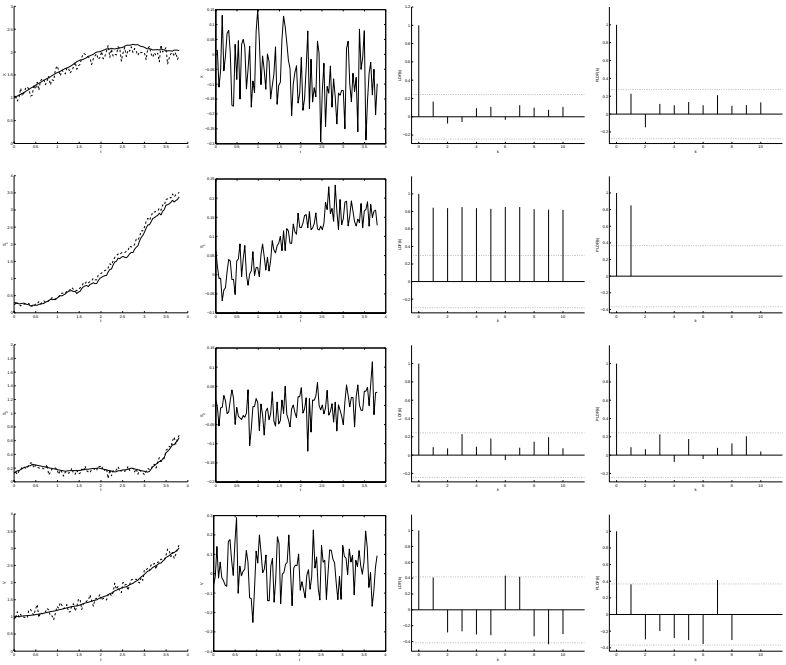


Figure 3.7. Pure simulation cross-validation residual analysis results for the model in (3.40) and (3.30) with parameters in Table 3.8 using the validation data set shown in Figure 3.2. Top-down: y_1, y_2, y_3 and y_4. Left-right: Pure simulation comparison (solid lines: Simulated values), residuals, LDF and PLDF.

3.3 A case with multiple deficiencies

To demonstrate the performance of the proposed grey-box modelling framework for a model with multiple deficiencies, the following example is considered.

Example 3.3 (A case with multiple deficiencies)
This example demonstrates the performance of the proposed grey-box modelling framework for a fed-batch fermentation process represented by a simulation model that describes growth of biomass and formation of a single product (penicillin) from a single substrate. The model is given as follows (Bajpai and Reuss, 1981):

$$\frac{dX}{dt} = \alpha(S,X)X - \frac{FX}{V} \tag{3.41}$$

$$\frac{dS}{dt} = -\frac{\alpha(S,X)X}{Y_X} - \frac{\theta(S)X}{Y_P} - M_X X + \frac{F\,(S_F - S)}{V} \tag{3.42}$$

$$\frac{dP}{dt} = \theta(S)X - KP - \frac{FP}{V} \tag{3.43}$$

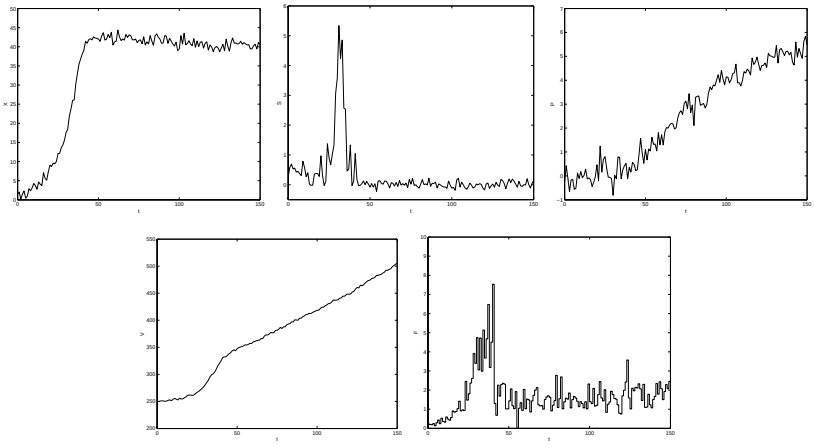


Figure 3.8. Data set no. 1 for Example 3.3. Top: X, S, P. Bottom: V, F.

$$\frac{dV}{dt} = F \tag{3.44}$$

for $t \in [t_0, t_f]$, where X ($\frac{g}{l}$) is the biomass concentration, S ($\frac{g}{l}$) is the substrate concentration, P ($\frac{g}{l}$) is the product concentration, V (l) is the reactor volume, F ($\frac{l}{h}$) is the feed flow rate, $Y_X = 0.47$ and $Y_P = 1.2$ are yield coefficients and $S_F = 400\frac{g}{l}$ is the substrate feed concentration. $M_X = 0.029h^{-1}$ represents a constant specific maintenance demand of the cells and K represents a constant first-order decay rate for the product. $t_0 = 0h$ and $t_f = 150h$ are initial and final times of a typical fed-batch run and $\alpha(S, X)$ (h^{-1}) and $\theta(S)$ (h^{-1}) are the biomass growth rate and the product formation rate respectively, i.e. (Bajpai and Reuss, 1981):

$$\begin{aligned} \alpha(S, X) &= \alpha_{\max} \frac{S}{S + K_1 X} \\ \theta(S) &= \theta_{\max} \frac{S}{K_{22} S^2 + S + K_{21}} \end{aligned} \tag{3.45}$$

where $\alpha_{\max} = 0.11h^{-1}$, $K_1 = 0.006$, $\theta_{\max} = 0.004h^{-1}$, $K_{21} = 0.0001\frac{g}{l}$ and $K_{22} = 10\frac{l}{g}$ are kinetic parameters. In order to generate data from this model by perturbing the feed flow rate along an appropriate trajectory, an optimal such trajectory is first determined by solving a productivity maximization problem equivalent to the one treated by Visser (1999). This problem can be stated as follows:

$$\max_{F(t)\ ,\ t \in [t_0, t_f]} P(t_f) \tag{3.46}$$

subject to the model equations and constraints on the maximum biomass and substrate concentrations and on the feed flow rate, using the initial conditions $X_0 = 1\frac{g}{l}$, $S_0 = 0.5\frac{g}{l}$ (Visser (1999) uses $0.2\frac{g}{l}$), $P_0 = 0\frac{g}{l}$ and $V_0 = 250l$. In other words, the problem is to determine the open loop feed flow rate trajectory that gives optimal productivity in terms of the product concentration at the end of a run.

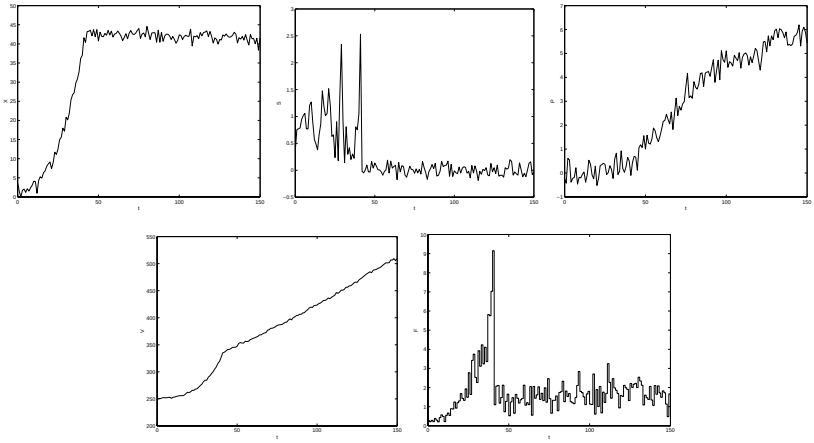


Figure 3.9. Data set no. 2 for Example 3.3. Top: X, S, P. Bottom: V, F.

The above maximization problem is solved in a manner similar to the one used by Visser (1999), and, by using perturbed versions of the resulting feed flow rate trajectory, two data sets (shown in Figures 3.8-3.9) are generated by means of stochastic simulation using the Euler scheme (see Example 2.2). For this purpose a re-formulated version of the model is applied, which has the following system equation:

$$d\begin{pmatrix} X \\ S \\ P \\ V \end{pmatrix} = \begin{pmatrix} \alpha(S,X)X - \frac{FX}{V} \\ -\frac{\alpha(S,X)X}{Y_X} - \frac{\theta(S)X}{Y_P} - M_X X + \frac{F(S_F - S)}{V} \\ \theta(S)X - KP - \frac{FP}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.47)$$

where $t \in [t_0, t_f]$, and the following measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ P \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 & 0 \\ 0 & S_{22} & 0 & 0 \\ 0 & 0 & S_{33} & 0 \\ 0 & 0 & 0 & S_{44} \end{bmatrix} \quad (3.48)$$

The parameter values applied are the deterministic parameter values mentioned above, the diffusion term parameter values $\sigma_{11} = \sigma_{22} = \sigma_{33} = \sigma_{44} = 0$ and the measurement noise term parameter values $S_{11} = 1$, $S_{22} = 0.01$, $S_{33} = 0.1$ and $S_{44} = 1$. A discretization time interval corresponding to $\frac{1}{150000}$ of t_f is used and every 100'th value is sampled (see Example 2.2) to give data sets containing 151 samples each.

Using the generated data sets, the performance of the grey-box modelling cycle and the corresponding algorithm for systematic iterative model improvement is now illustrated by assuming that an initial model structure corresponding to (3.47)-(3.48) is available, where the true structure of the biomass growth rate $\alpha(S, X)$ as well as the true structure of the product formation rate $\theta(S)$ are unknown. In other words, it is

assumed that Steps 1 and 2 of the algorithm, which deal with derivation of an ODE model from first engineering principles and translation of this model into a continuous-discrete stochastic state space model with a diagonally parameterized diffusion term, have been completed to yield a model with the following system equation:

$$d\begin{pmatrix} X \\ S \\ P \\ V \end{pmatrix} = \begin{pmatrix} \alpha X - \frac{FX}{V} \\ -\frac{\alpha X}{Y_X} - \frac{\theta X}{Y_P} - M_X X + \frac{F(S_F - S)}{V} \\ \theta X - KP - \frac{FP}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.49)$$

where $t \in [t_0, t_f]$, and where, because the true structures of the biomass growth rate and the product formation rate are unknown, constant rates α and θ have been assumed. The measurement equation of the model is equivalent to (3.48). In Step 3 of the algorithm, the unknown parameters of the model are estimated using **CTSM** and the data set in Figure 3.8, which gives the results shown in Table 3.9. To evaluate the quality of the resulting model in terms of its prediction capabilities, cross-validation residual analysis is performed in Step 4, and, since the intended purpose of the model is assumed to be application for subsequent state estimation and optimal control, which requires a model with good long-term prediction capabilities, only pure simulation residual analysis is performed, cf. Figure 3.10. The results of this analysis show that the model has very poor pure simulation capabilities and thus falsify the model for the purpose of optimal control in Step 5, which means that the model development procedure implied by the grey-box modelling cycle must be repeated by re-formulating the model. Step 6 of the algorithm, which deals with pinpointing of model deficiencies, is therefore applied. Table 3.9 includes t-scores for performing marginal tests for insignificance of the individual parameters, and, on a 5% level, these show that, of the parameters of the diffusion term, only σ_{44} is insignificant.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.4894E+00	1.4340E+00	1.0387	No
S_0	2.5616E-01	1.2743E+00	0.2010	No
P_0	5.3776E-11	1.8798E-08	0.0029	No
V_0	2.5009E+02	7.5880E-02	3295.9283	Yes
α	6.9525E-03	2.4324E-03	2.8583	Yes
θ	1.8263E-03	2.9069E-04	6.2828	Yes
M_X	2.8732E-02	5.7193E-03	5.0236	Yes
K	5.1610E-03	3.3556E-03	1.5380	No
σ_{11}	1.1527E+00	1.0547E-01	10.9296	Yes
σ_{22}	1.3718E+00	8.7977E-02	15.5927	Yes
σ_{33}	5.8930E-02	2.2987E-02	2.5636	Yes
σ_{44}	7.5747E-08	7.6491E-06	0.0099	No
S_{11}	2.9803E-01	1.2588E-01	2.3675	Yes
S_{22}	2.5004E-15	7.4715E-13	0.0033	No
S_{33}	8.6803E-02	1.3321E-02	6.5164	Yes
S_{44}	9.0304E-01	9.6043E-02	9.4025	Yes

Table 3.9. Estimation results. Model in (3.49) and (3.48) - data from Figure 3.8.

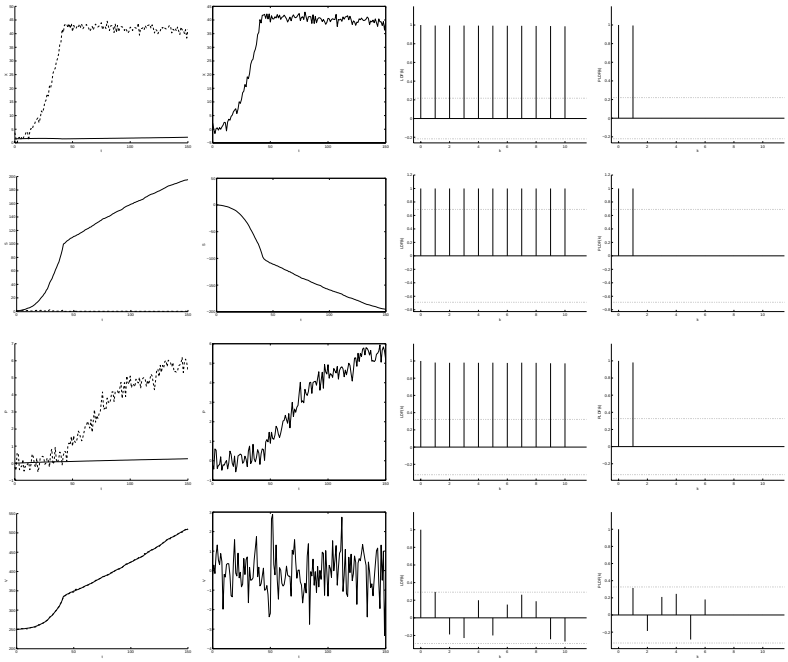
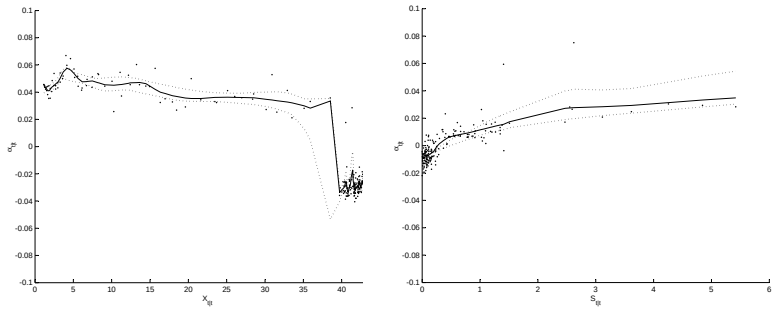


Figure 3.10. Pure simulation cross-validation residual analysis results for the model in (3.49) and (3.48) with parameters in Table 3.9 using the validation data set shown in Figure 3.9. Top-down: y_1, y_2, y_3 and y_4. Left-right: Pure simulation comparison (solid lines: Simulated values), residuals, LDF and PLDF.

The fact that the remaining parameters of the diffusion term are all significant, indicates that the corresponding elements of the drift term may be incorrect. These elements all depend on α and θ, which means that these are possible model deficiency suspects. Because the σ_{11} and σ_{22} parameters of the diffusion term, which correspond to α-dependent elements of the drift term, are more significant than σ_{33}, which corresponds to a purely θ-dependent element of the drift term, α is investigated first by re-formulating the model with α as an additional state variable as follows:

$$d\begin{pmatrix} X \\ S \\ P \\ V \\ \alpha \end{pmatrix} = \begin{pmatrix} \alpha X - \frac{FX}{V} \\ -\frac{\alpha X}{Y_X} - \frac{\theta X}{Y_P} - M_X X + \frac{F(S_F - S)}{V} \\ \theta X - KP - \frac{FP}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 & 0 \\ 0 & 0 & 0 & \sigma_{44} & 0 \\ 0 & 0 & 0 & 0 & \sigma_{55} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.50)$$

where $t \in [t_0, t_f]$, and where the last element of the drift term is zero, because α has been assumed to be constant. The measurement equation corresponding to the above system equation remains equivalent to (3.48). Estimating the unknown parameters



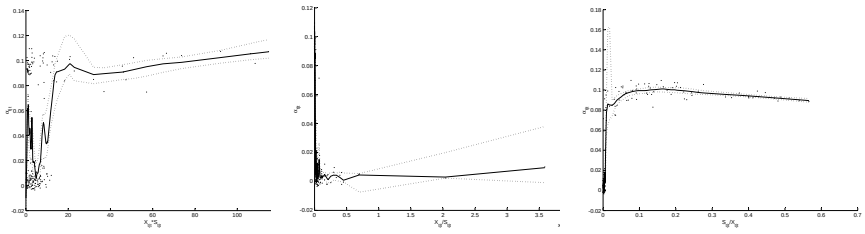
(a) $\hat{\alpha}_{k|k}$ vs. $\hat{X}_{k|k}$. (b) $\hat{\alpha}_{k|k}$ vs. $\hat{S}_{k|k}$.

Figure 3.11. Partial dependence plots of $\hat{\alpha}_{k|k}$ vs. $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$ obtained by applying additive model fitting using locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths determined using 5-fold cross-validation). Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals computed from 1000 replicates (see Appendix C for details).

of this model using **CTSM** and the same data set as before, gives the results shown in Table 3.10, and inspection of the t-scores for marginal tests for insignificance now show that, of the parameters of the diffusion term, only σ_{33} and σ_{55} are significant.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.1669E+00	2.2699E-01	5.1409	Yes
S_0	4.6705E-01	9.6849E-02	4.8225	Yes
P_0	2.3566E-10	1.3486E-06	0.0002	No
V_0	2.5011E+02	7.8001E-02	3206.4513	Yes
α_0	9.3196E-02	2.0777E-02	4.4855	Yes
θ	1.8418E-03	3.0702E-04	5.9990	Yes
M_X	2.7945E-02	2.8819E-04	96.9703	Yes
K	5.2749E-03	3.5005E-03	1.5069	No
σ_{11}	4.7313E-25	3.1238E-21	0.0002	No
σ_{22}	2.3911E-21	4.7886E-17	0.0000	No
σ_{33}	5.9890E-02	2.4851E-02	2.4099	Yes
σ_{44}	1.1942E-13	3.3076E-10	0.0004	No
σ_{55}	6.0596E-03	8.7587E-04	6.9184	Yes
S_{11}	7.8432E-01	8.8697E-02	8.8427	Yes
S_{22}	6.4526E-02	1.4364E-02	4.4922	Yes
S_{33}	9.0063E-02	1.3188E-02	6.8290	Yes
S_{44}	9.1818E-01	1.0553E-01	8.7008	Yes

Table 3.10. Estimation results. Model in (3.50) and (3.48) - data from Figure 3.8.



(a) $\hat{\alpha}_{k|k}$ vs. $\hat{X}_{k|k}\hat{S}_{k|k}$. (b) $\hat{\alpha}_{k|k}$ vs. $\hat{X}_{k|k}/\hat{S}_{k|k}$. (c) $\hat{\alpha}_{k|k}$ vs. $\hat{S}_{k|k}/\hat{X}_{k|k}$.

Figure 3.12. Independent kernel estimates of the dependence between $\hat{\alpha}_{k|k}$ and $\hat{X}_{k|k}\hat{S}_{k|k}$, $\hat{X}_{k|k}/\hat{S}_{k|k}$ and $\hat{S}_{k|k}/\hat{X}_{k|k}$ obtained by applying locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths obtained with 5-fold cross-validation). Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals computed from 1000 replicates (see Appendix C for details).

The fact that σ_{55} is significant, indicates that there is substantial variation in α and thus confirms the suspicion that α is deficient. Moving to Step 7 of the algorithm, nonparametric modelling can now be applied to determine how to improve the model.

Using the re-formulated model in (3.50) and (3.48) and the parameter estimates in Table 3.10, state estimates $\hat{X}_{k|k}$, $\hat{S}_{k|k}$, $\hat{P}_{k|k}$, $\hat{V}_{k|k}$, $\hat{\alpha}_{k|k}$, $k = 0, \ldots, N$, are computed with **CTSM** from the data sets shown in Figures 3.8-3.9 and an additive model is fitted to reveal the true structure of the function describing α by means of estimates of functional relations between α and the original state variables. It is assumed that α does not depend on P and V, so only functional relations between $\hat{\alpha}_{k|k}$ and $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$ (with negative values removed) are estimated, which gives the results shown in Figure 3.11. These plots indicate that $\hat{\alpha}_{k|k}$ depends slightly on both $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$, and because of the apparent dependence on more than one variable, further investigations are needed to rule out the possibility that this is caused by an actual dependence on e.g. the product of these variables or a fraction between them.

Figure 3.12 shows independent kernel estimates of the dependence between $\hat{\alpha}_{k|k}$ and the product $\hat{X}_{k|k}\hat{S}_{k|k}$ and the fractions $\hat{X}_{k|k}/\hat{S}_{k|k}$ and $\hat{S}_{k|k}/\hat{X}_{k|k}$ respectively. These plots show that neither $\hat{X}_{k|k}\hat{S}_{k|k}$ nor $\hat{X}_{k|k}/\hat{S}_{k|k}$ describe the variations in $\hat{\alpha}_{k|k}$ particularly well, whereas $\hat{S}_{k|k}/\hat{X}_{k|k}$ provides a much better description. More specifically, the functional relation revealed in Figure 3.12c is indicative of a functionality that can be described by an expression of the Monod type in the variable S/X, i.e.:

$$\alpha(\frac{S}{X}) = \alpha_{\max}\frac{\frac{S}{X}}{\frac{S}{X} + K_1} \tag{3.51}$$

which is equivalent to the following expression in the original variables S and X:

$$\alpha(S, X) = \alpha_{\max}\frac{S}{S + K_1 X} \tag{3.52}$$

In Step 8 of the algorithm, it is therefore reasonable to re-formulate the model by replacing the assumption of constant α with an assumption of α being described by this expression, which yields a model with the following system equation:

$$d\begin{pmatrix} X \\ S \\ P \\ V \end{pmatrix} = \begin{pmatrix} \alpha(S,X)X - \frac{FX}{V} \\ -\frac{\alpha(S,X)X}{Y_X} - \frac{\theta X}{Y_P} - M_X X + \frac{F(S_F - S)}{V} \\ \theta X - KP - \frac{FP}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.53)$$

where $t \in [t_0, t_f]$. The measurement equation remains equivalent to (3.48). Returning to Step 3 of the algorithm, the unknown parameters of the new model are estimated using **CTSM** and the data set in Figure 3.8, which gives the results shown in Table 3.11, and in Step 4 the quality of the resulting model is evaluated by performing cross-validation residual analysis, cf. Figure 3.13. The results of this analysis show that the new model has significantly better pure simulation capabilities than the previously analyzed model. More specifically, the y_1 and y_4 residuals can be regarded as white noise, and the y_2 and y_3 pure simulation comparisons are much better than with the previously analyzed model. However, there seems to be a little non-random variation still left in the y_2 and y_3 residuals, and, depending on the specific degree of accuracy required, this model may therefore also be falsified for the purpose of optimal control in Step 5 of the algorithm. Assuming that this is the case, the model development procedure must be repeated by re-formulating the model, and Step 6, which deals with pinpointing of model deficiencies, is therefore applied. The t-scores included in Table 3.11 show that one of the parameters of the diffusion term is significant, i.e. σ_{33}, and this indicates that the corresponding element of the drift term may be incorrect. This element depends on θ, which is thus a candidate for being de-

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.8702E-01	1.4390E-02	68.5902	Yes
S_0	4.6596E-01	3.7383E-02	12.4646	Yes
P_0	7.4709E-09	3.2743E-07	0.0228	No
V_0	2.5009E+02	7.6073E-02	3287.4706	Yes
$\alpha_{\max}$	1.0968E-01	4.5201E-04	242.6492	Yes
K_1	5.8609E-03	4.6530E-04	12.5960	Yes
θ	1.8030E-03	2.9919E-04	6.0263	Yes
M_X	2.7947E-02	2.7507E-04	101.6025	Yes
K	4.9048E-03	3.6378E-03	1.3483	No
σ_{11}	1.2391E-08	3.4938E-07	0.0355	No
σ_{22}	5.9098E-07	1.2459E-05	0.0474	No
σ_{33}	6.0986E-02	2.4815E-02	2.4576	Yes
σ_{44}	1.1148E-09	3.6180E-08	0.0308	No
S_{11}	7.9785E-01	9.7841E-02	8.1546	Yes
S_{22}	9.1256E-03	1.0735E-03	8.5006	Yes
S_{33}	9.0496E-02	1.4242E-02	6.3540	Yes
S_{44}	9.3088E-01	1.0865E-01	8.5679	Yes

Table 3.11. Estimation results. Model in (3.53) and (3.48) - data from Figure 3.8.

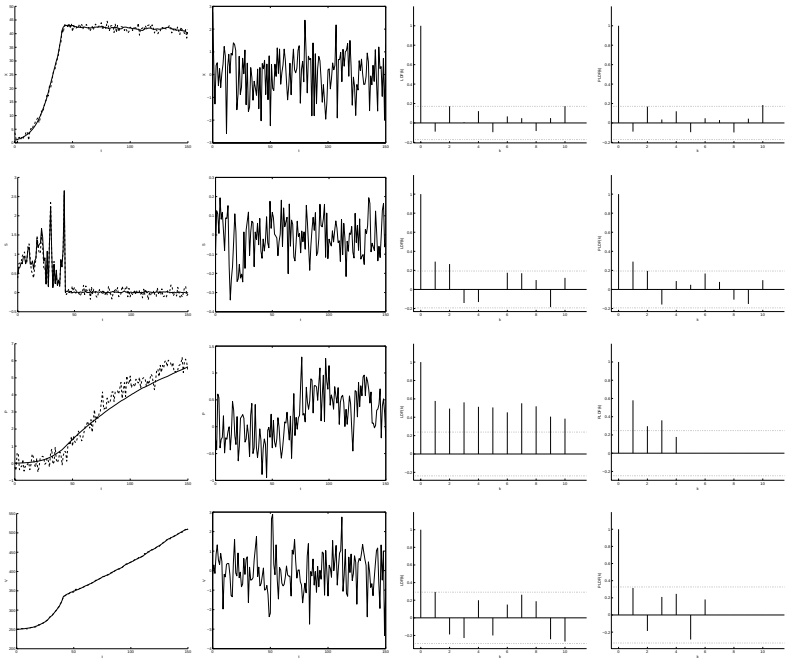


Figure 3.13. Pure simulation cross-validation residual analysis results for the model in (3.53) and (3.48) with parameters in Table 3.11 using the validation data set shown in Figure 3.9. Top-down: y_1, y_2, y_3 and y_4. Left-right: Pure simulation comparison (solid lines: Simulated values), residuals, LDF and PLDF.

ficient. That this may be the case is supported by the above residual analysis results, which show that the y_2 and y_3 residuals, which correspond to state variables with θ-dependent drift term elements, still contain a little non-random variation. However, to avoid jumping to conclusions, the suspicion that θ is deficient is investigated further by re-formulating the model with θ as an additional state variable as follows:

$$d\begin{pmatrix} X \\ S \\ P \\ V \\ \theta \end{pmatrix} = \begin{pmatrix} \alpha(S,X)X - \frac{FX}{V} \\ -\frac{\alpha(S,X)X}{Y_X} - \frac{\theta X}{Y_P} - M_X X + \frac{F(S_F - S)}{V} \\ \theta X - KP - \frac{FP}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 & 0 \\ 0 & 0 & 0 & \sigma_{44} & 0 \\ 0 & 0 & 0 & 0 & \sigma_{55} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.54)$$

where $t \in [t_0, t_f]$, and where the last element of the drift term is zero, because θ has been assumed to be constant. The measurement equation corresponding to the above system equation remains equivalent to (3.48). Estimating the unknown parameters of this model using **CTSM** and the same data set as before, gives the results shown in Table 3.12, and inspection of the t-scores for marginal tests for insignificance now

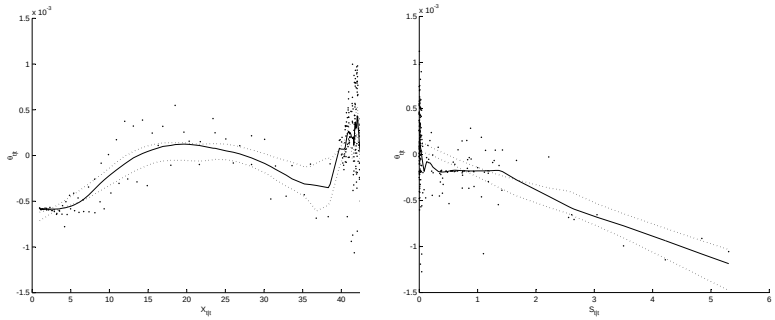
show that, of the parameters of the diffusion term, only σ_{55} is significant. This indicates that there is substantial variation in θ and thus confirms the suspicion that θ is deficient. Moving to Step 7 of the algorithm, nonparametric modelling can now be applied in an attempt to determine how to improve the model, if this is possible.

Using the re-formulated model in (3.54) and (3.48) and the parameter estimates in Table 3.12, state estimates $\hat{X}_{k|k}$, $\hat{S}_{k|k}$, $\hat{P}_{k|k}$, $\hat{V}_{k|k}$, $\hat{\theta}_{k|k}$, $k = 0, \ldots, N$, are computed with **CTSM** from the data sets shown in Figures 3.8-3.9 and an additive model is fitted to reveal the true structure of the function describing θ by means of estimates of functional relations between θ and the original state variables. It is assumed that θ does not depend on P and V, so only functional relations between $\hat{\theta}_{k|k}$ and $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$ (with negative values removed) are estimated, which gives the results shown in Figure 3.14. Apart from a slightly decreasing tendency in the plot of $\hat{\theta}_{k|k}$ vs. $\hat{S}_{k|k}$, these plots do not provide much useful information due to the low degree of variation in $\hat{\theta}_{k|k}$ ($\hat{\theta}_{k|k}$ also seems to depend on $\hat{X}_{k|k}$, but in a rather complicated manner, and further investigations indicate that the apparent dependence on more than one variable does not seem to be caused by an actual dependence on e.g. the product of these variables or a fraction between them). Nevertheless, this tendency may be interpreted as an indication of inhibition of product formation at high substrate concentrations, which makes it reasonable to replace the assumption of constant θ with an assumption of θ being a function of S that can be described with Monod kinetics (this may be percieved as conjecture but is supported by the fact that bioprocesses are often governed by kinetics of this type) and substrate inhibition, i.e.:

$$\theta(S) = \theta_{\max} \frac{S}{K_{22}S^2 + S + K_{21}} \tag{3.55}$$

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.8971E-01	1.4320E-02	69.1130	Yes
S_0	4.6288E-01	3.6571E-02	12.6572	Yes
P_0	4.7897E-28	8.0233E-25	0.0006	No
V_0	2.5009E+02	8.1135E-02	3082.4156	Yes
θ_0	9.8568E-04	5.3409E-04	1.8455	No
$\alpha_{\max}$	1.0966E-01	4.1399E-04	264.8811	Yes
K_1	5.8465E-03	4.1862E-04	13.9659	Yes
M_X	2.7793E-02	3.0794E-04	90.2557	Yes
K	7.8619E-03	5.2358E-03	1.5016	No
σ_{11}	1.0126E-15	7.9983E-13	0.0013	No
σ_{22}	4.2047E-07	7.1777E-05	0.0059	No
σ_{33}	1.4257E-04	1.5702E-03	0.0908	No
σ_{44}	6.5830E-06	5.5897E-04	0.0118	No
σ_{55}	9.6323E-05	3.7177E-05	2.5909	Yes
S_{11}	7.9247E-01	8.6839E-02	9.1257	Yes
S_{22}	9.1355E-03	9.7903E-04	9.3312	Yes
S_{33}	1.0249E-01	1.1763E-02	8.7128	Yes
S_{44}	9.2910E-01	1.0127E-01	9.1743	Yes

Table 3.12. Estimation results. Model in (3.54) and (3.48) - data from Figure 3.8.



(a) $\hat{\theta}_{k|k}$ vs. $\hat{X}_{k|k}$. (b) $\hat{\theta}_{k|k}$ vs. $\hat{S}_{k|k}$.

Figure 3.14. Partial dependence plots of $\hat{\theta}_{k|k}$ vs. $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$ obtained by applying additive model fitting using locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths determined using 5-fold cross-validation). Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals computed from 1000 replicates (see Appendix C for details).

This replacement of assumptions yields a model with the following system equation:

$$d\begin{pmatrix} X \\ S \\ P \\ V \end{pmatrix} = \begin{pmatrix} \alpha(S,X)X - \frac{FX}{V} \\ -\frac{\alpha(S,X)X}{Y_X} - \frac{\theta(S)X}{Y_P} - M_X X + \frac{F(S_F - S)}{V} \\ \theta(S)X - KP - \frac{FP}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \quad (3.56)$$

where $t \in [t_0, t_f]$. The measurement equation remains equivalent to (3.48). Returning to Step 3 of the algorithm, the unknown parameters of the new model are estimated using **CTSM** and the data set in Figure 3.8, which gives the results shown in Table 3.13, and in Step 4 the quality of the resulting model is evaluated by performing cross-validation residual analysis, cf. Figure 3.15. The results of this analysis show that the model has better pure simulation capabilities than the previously analyzed model. In particular, the y_3 pure simulation comparison has improved. Nevertheless, there still seems to be a little non-random variation left in the y_2 and y_3 residuals, and depending on the specific degree of accuracy required, the new model may therefore also be falsified for the purpose of optimal control in Step 5, meaning that the model development procedure must be repeated by re-formulating the model, but this is assumed not to be the case. Furthermore, all information available in the data set used for estimation has been exhausted in the context of the proposed grey-box modelling framework, because a model has been developed where the diffusion term

is insignificant[2], which means that model deficiencies can no longer be systematically pinpointed. Moreover, the true model in (3.47)-(3.48) has been recovered. ■

The above example demonstrates the performance of the proposed grey-box modelling framework for a model with multiple deficiencies. In particular, the example demonstrates that, if a model has multiple deficiencies, these can be repaired one at a time by applying the methods of the proposed grey-box modelling cycle and the corresponding algorithm for systematic iterative model improvement in a successive manner. Furthermore, the example demonstrates that a deficiency caused by an incorrectly modelled function of more than one variable can sometimes be repaired in a single step, if, unlike in the previous example, this function is a simple function of e.g. the product of these variables or a fraction between them. However, the example also demonstrates that, if the degree of variation in key variables is insufficient, systematic model development may not be possible. In other words, the example demonstrates that the performance of the proposed framework is limited by the information content of the data sets used for model development. This stresses the need for developing methods for experimental design that can be applied along with

[2]Inspection of the t-scores for marginal tests for insignificance (Table 3.13) suggest that, on a 5% level, there are no significant parameters in the diffusion term, which is confirmed by a test for simultaneous insignificance based on Wald's W-statistic. A final calibration of the remaining model parameters should therefore ideally be performed at this stage, using an estimation method that emphasizes the pure simulation capabilities of the model.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.8164E-01	1.3211E-02	74.3033	Yes
S_0	4.5540E-01	3.6173E-02	12.5896	Yes
P_0	6.9569E-26	1.1431E-21	0.0001	No
V_0	2.5009E+02	8.3471E-02	2996.1921	Yes
$\alpha_{\max}$	1.0998E-01	4.0924E-04	268.7277	Yes
K_1	5.6799E-03	4.2219E-04	13.4536	Yes
$\theta_{\max}$	9.9755E-03	8.4511E-05	118.0383	Yes
K_{21}	9.9640E-03	1.3710E-04	72.6766	Yes
K_{22}	1.6124E+01	1.4822E+00	10.8786	Yes
M_X	2.7717E-02	1.3169E-04	210.4657	Yes
K	7.7384E-03	8.3263E-04	9.2939	Yes
σ_{11}	6.8050E-17	6.4282E-13	0.0001	No
σ_{22}	8.8487E-09	2.7909E-05	0.0003	No
σ_{33}	1.4428E-06	2.0700E-03	0.0007	No
σ_{44}	1.6264E-06	2.2635E-03	0.0007	No
S_{11}	7.9829E-01	8.8955E-02	8.9741	Yes
S_{22}	9.1150E-03	9.9032E-04	9.2041	Yes
S_{33}	1.4798E-01	1.7056E-02	8.6761	Yes
S_{44}	9.2911E-01	1.0322E-01	9.0014	Yes

Table 3.13. Estimation results. Model in (3.56) and (3.48) - data from Figure 3.8.

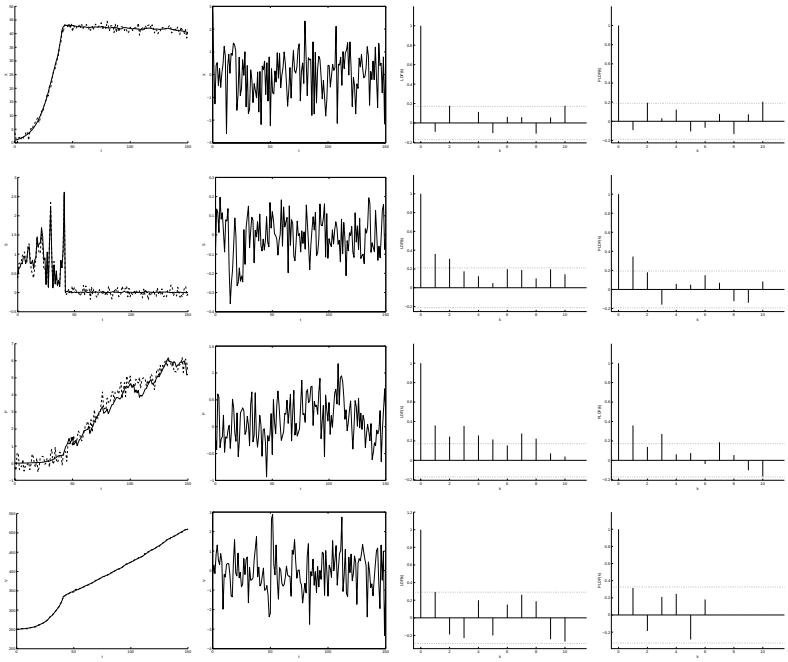


Figure 3.15. Pure simulation cross-validation residual analysis results for the model in (3.56) and (3.48) with parameters in Table 3.13 using the validation data set shown in Figure 3.9. Top-down: y_1, y_2, y_3 and y_4. Left-right: Pure simulation comparison (solid lines: Simulated values), residuals, LDF and PLDF.

the proposed grey-box modelling framework to ensure that a maximum of information is obtained, given the specific circumstances, in terms of operational limitations, under which experiments can be performed for a given fed-batch process, but this is outside the scope of the work presented in this thesis.

4

Conclusion

The primary focus of the work presented in this thesis has been on modelling of fed-batch processes for the purpose of state estimation and optimal control.

The motivation for focusing on this issue have been the shortcomings of present industrial approaches to operation of fed-batch processes with respect to achieving uniform operation and optimal productivity and the resulting need for development of an appropriate model-based approach to automatic operation capable of achieving these goals. A number of requirements for such an approach have been listed and a review of various approaches reported in literature has been given along with a discussion of their merits with respect to meeting these requirements. This review has indicated that an approach incorporating continuous-discrete stochastic state space models may be particularly advantageous, because such models combine the strengths of first engineering principles models and data-driven models, neither of which seem fully adequate for modelling fed-batch processes for the purpose of achieving uniform operation and optimal productivity. In particular, developing first engineering principles models is time-consuming, because few systematic methods are available for making inferences about the proper structure of such models, which can seldom be determined completely from prior physical knowledge. Furthermore, the parameters of such models can only be estimated from experimental data by using OE estimation methods, which has been demonstrated through a simple comparison to give more biased and less reproducible results in the presence of significant process noise than the PE estimation methods, which can be applied for data-driven models. On the other hand, data-driven models, for which systematic methods for structural identification are also available, are not as intuitively appealing as first engineering principles models in terms of providing a consistent and physically meaningful system description. Continuous-discrete stochastic state space models combine the strengths of both model types by allowing first engineering principles to be applied and prior physical knowledge to be incorporated, while providing a decomposition of the noise affecting the system into a process noise term and a measurement noise term, which facilitates PE estimation and subsequent application of powerful statistical tools.

Based on continuous-discrete stochastic state space models, the main features of an overall framework for fed-batch process modelling, state estimation and

optimal control have been established. This framework incorporates modelling as well as experimental design and state estimation and optimal control, but in the work presented in this thesis attention has been restricted to the modelling part, to facilitate which a grey-box modelling framework has been proposed.

This framework is based on a grey-box modelling cycle, the idea of which is to facilitate the development of models of fed-batch processes for the purpose of state estimation and optimal control. The modelling cycle comprises six different tasks: Model (re)formulation, where the idea is to use first engineering principles and all other relevant prior physical knowledge to construct an initial continuous-discrete stochastic state space model; parameter estimation, where the idea is to estimate the parameters of this model from available experimental data; residual analysis, where the idea is to perform cross-validation residual analysis to obtain information about the quality of the resulting model; model falsification or unfalsification, where the idea is to use this information to determine if the model is sufficiently accurate to be used for state estimation and optimal control; statistical tests, where, if the model is falsified for this purpose with respect to the available information, the idea is to pinpoint deficiencies within the model, if this is possible; and nonparametric modelling, where the idea is to determine how to repair these deficiencies by altering the model when afterwards returning to the model (re)formulation task to complete the cycle.

The grey-box modelling cycle is the main result of the work presented in this thesis, and much emphasis has been put on developing simple methods and tools to facilitate its individual tasks. A significant result in this regard is the extension of an existing parameter estimation method for continuous-discrete stochastic state space models by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) to make it more readily applicable to models of fed-batch processes and the implementation of this method in a computer program called **CTSM**. As part of these developments, the inability of the original estimation method to handle models with singular Jacobians has been remedied and the method has been extended to allow estimation with multiple independent sets of experimental data and to handle missing observations in a much more appropriate way. With respect to **CTSM**, which is based on a similar program by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) called CTLSM, the program has been equipped with a graphical user interface for ease of use, and for the purpose of computational efficiency the binary code of the program has been optimized and prepared for shared memory parallel computing. An important result with respect to this program is that it has proven superior, both in terms of quality of estimates and in terms of reproducibility, to another program implementing a similar estimation method by Bohlin and Graebe (1995) and Bohlin (2001). In particular, more accurate and more consistent estimates of the parameters of the diffusion term can be obtained, which is important in the context of the proposed grey-box modelling framework. A number of additional tools that facilitate other tasks within the grey-box modelling cycle have also been developed and implemented in MATLAB, and based on all

of the individual tasks of the modelling cycle a grey-box modelling algorithm that facilitates systematic iterative model improvement has been presented.

A key feature of the methodology provided by the grey-box modelling cycle and the corresponding algorithm is that it facilitates pinpointing of model deficiencies based on information extracted from experimental data and subsequently allows the structural origin of these deficiencies to be uncovered as well to provide guidelines for model improvement. The procedure for pinpointing model deficiencies is based on the fact that estimation of the parameters of the diffusion term provides a measure of the uncertainty of the corresponding drift term. This means that, if a diagonal parameterization is used, the uncertainty of a particular element of the drift term can be assessed, and, by proper reformulation of the model, suspicions of deficiencies in particular parts of such terms, e.g. parts describing dynamic phenomena such as reaction rates and heat and mass transfer rates, can be confirmed as well. Once such specific deficiencies have been confirmed, the same model can be used to obtain state estimates, on the basis of which nonparametric estimates of unknown or incorrectly modelled functional relations can be obtained and visualized, whereby the structural origin of these deficiencies can be uncovered and the model subsequently improved. This is a very powerful feature not shared by other approaches to grey-box modelling reported in literature, e.g. the approach by Bohlin and Graebe (1995) and Bohlin (2001), which relies solely on the model maker to determine how to improve the model. In this particular sense, the methodology proposed here is therefore more systematic, which is a key result.

The performance of the proposed methodology has been demonstrated through a number of application examples, the most simple of which has demonstrated that, in a case where all state variables are measured directly, a deficiency caused by an incorrectly modelled function of a single state variable can easily be pinpointed and its structural origin subsequently uncovered. A similar example, where the particular state variable occuring in the incorrectly modelled function causing the deficiency is not measured, has demonstrated that the same is also possible in cases where all state variables cannot be measured directly. Additional examples have demonstrated that the proposed methodology allows deficiencies caused by incorrectly modelled functions of more than one state variable to be handled as well, either in a single step, which may be possible if the incorrectly modelled function depends on e.g. the product of these variables or a fraction between them, or in a stepwise manner. Finally, it has been demonstrated that the methodology can be successfully applied in cases with multiple deficiencies as well. However, the application examples have also demonstrated that the proposed methodology has certain limitations.

Like other approaches to grey-box modelling, the performance of the proposed methodology is limited by the quality and amount of available prior physical knowledge and experimental data. More specifically, there may be insufficient prior physical knowledge available to establish an initial model structure, in which case it may not be worthwhile to use this approach as opposed to a

data-driven modelling approach. With respect to the available experimental data, it may be insufficiently informative or the available measurements may render certain subsets of the state variables of the system unobservable, in which case parameter identifiability may be seriously affected. The procedure for pinpointing model deficiencies relies on estimates of the parameters of the diffusion term and the procedure for subsequently uncovering the structural origin of these deficiencies requires that the state variables of the system are observable, which means that the reliability of these procedures may be affected as well. Another obvious limitation with regards to these procedures is that the model maker may be unable to select specific phenomena for further investigation when model deficiencies have been indicated, which is an important prerequisite for using these procedures. In other words, although much less reliant on, the proposed methodology is not independent of the model maker.

An important question with respect to the proposed methodology is the matter of whether or not a guarantee of convergence can be given. More specifically, assuming that a "true" model exists, where all state variables are observable, and that the available experimental data is sufficiently informative to ensure that all parameters are identifiable, will the grey-box modelling algorithm then converge to yield the "true" model? In the general case, no rigorous proof of such convergence exists, but the application examples have demonstrated that the algorithm may in fact converge for certain simple systems. In any case, the proposed methodology can be applied to facilitate faster model development.

In conclusion, the work presented in this thesis has resulted in the development of a systematic grey-box modelling framework, which, through novel procedures for pinpointing and subsequently uncovering the structural origin of model deficiencies, facilitates the development of fed-batch process models which are suitable for subsequent state estimation and optimal control with the aim of achieving uniform operation and optimal productivity. As an additional result, a generalized version of the grey-box modelling framework, which can be applied to model a variety of systems for different purposes, has been developed.

5

Suggestions for future work

During the course of the work presented in this thesis a number of related problems have presented themselves, the treatment of which has been outside the scope of the work. Some of the most important of these are summarized in the following in the form of a number of possible topics for future work.

A very important such topic relates to the relaxation of the assumption made in Chapter 1 concerning additional implicit algebraic equations. This is clearly not a valid assumption in many practical cases and efforts should be made to extend the proposed grey-box modelling framework to be able to handle models with such equations as well, preferably in a way that allows the uncertainty of these equations to be assessed in order to be able to detect deficiencies in these as well. This is, however, not an easy task, as it is believed to require the use of stochastic differential algebraic equations (SDAE's), the theory of which is not very well developed, particularly not with respect to the associated filtering problem that must be solved in order to apply a parameter estimation method similar to the EKF-based method used in the work presented in this thesis.

Being a part of the overall framework for fed-batch process modelling, state estimation and optimal control established in Chapter 1 but otherwise outside the scope of the work presented here, experimental design is an obvious topic for future work. This is emphasized by the fact that the EKF-based method used for estimating the parameters of the model and the procedures for pinpointing and subsequently uncovering the structural origin of model deficiencies are all highly dependent on the quality and amount of available experimental data. To be more specific, efforts should be made to develop a systematic approach to the design of identification experiments, which ensures that sufficient information is obtained for the proposed grey-box modelling framework to be applicable. Considering the fact that the models being developed are to be used for subsequent state estimation and optimal control, where the latter requires good long-term prediction capabilities, it is evident that such an approach must ensure that data covering wide ranges of state space is obtained, but it should also reflect the fact that experiments on industrial scale processes are often expensive and should hence aim to minimize the amount of experimentation needed to obtain sufficient information. In this regard, it may be worthwhile to investigate whether using one normal batch (where operation is regular) and

one faulty batch (where something goes wrong and operation is irregular) of standard operational data provides sufficient information, the idea being that, by using one of each, a relatively wide range of state space is covered.

Likewise being a part of the overall framework established in Chapter 1 but otherwise outside the scope of the work presented here, another obvious topic for future work is the development of specific methods for optimal control with simultaneous state estimation based on continuous-discrete stochastic state space models. Such a method should be able to handle operational limitations such as state and input variable constraints, for which reason MPC is an obvious candidate, perhaps with simultaneous state estimation based on the EKF, because of the possibility of using optimal values for the parameters of the diffusion term and the measurement noise term provided by the likewise EKF-based parameter estimation method used in the work presented here. Alternatively, a method based on stochastic dynamic programming could be developed, which would allow the uncertainty implied by a possibly significant diffusion term to be handled in an appropriate way. This is, however, less straightforward.

Appendices

A

CTSM

In this appendix a complete mathematical outline of the algorithms of the computer program **CTSM** is given. **CTSM** is an abbreviation of **C**ontinuous **T**ime **S**tochastic **M**odelling and is based on a similar computer program by Madsen and Melgaard (1991) and Melgaard and Madsen (1993) called CTLSM.

CTSM provides features for parameter estimation in continuous-discrete stochastic state space models and, by allowing uncertainty information to be computed and validation data to be generated, the program also facilitates a number of other tasks within the grey-box modelling cycle described in Chapter 2.

A.1 Parameter estimation

The primary feature in **CTSM** is estimation of parameters in continuous-discrete stochastic state space models on the basis of experimental data.

A.1.1 Model structures

CTSM differentiates between three different model structures for continuous-discrete stochastic state space models as outlined in the following.

A.1.1.1 The nonlinear model

The most general of these model structures is the *nonlinear* (NL) model, which can be described by the following equations:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{A.1}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{A.2}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a vector of state variables, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is a vector of input variables, $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is a vector of output variables, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ and $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ are nonlinear functions, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

A.1.1.2 The linear time-varying model

A special case of the nonlinear model is the *linear time-varying* (LTV) model, which can be described by the following equations:

$$d\boldsymbol{x}_t = (\boldsymbol{A}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})\boldsymbol{x}_t + \boldsymbol{B}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})\boldsymbol{u}_t)\, dt + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta}) d\boldsymbol{\omega}_t \tag{A.3}$$

$$\boldsymbol{y}_k = \boldsymbol{C}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})\boldsymbol{x}_k + \boldsymbol{D}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta})\boldsymbol{u}_k + \boldsymbol{e}_k \tag{A.4}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a state vector, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is an input vector, $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is an output vector, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{A}(\cdot) \in \mathbb{R}^{n\times n}$, $\boldsymbol{B}(\cdot) \in \mathbb{R}^{n\times m}$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n\times n}$, $\boldsymbol{C}(\cdot) \in \mathbb{R}^{l\times n}$ and $\boldsymbol{D}(\cdot) \in \mathbb{R}^{l\times m}$ are nonlinear functions, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

A.1.1.3 The linear time-invariant model

A special case of the linear time-varying model is the *linear time-invariant* (LTI) model, which can be described by the following equations:

$$d\boldsymbol{x}_t = (\boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{x}_t + \boldsymbol{B}(\boldsymbol{\theta})\boldsymbol{u}_t)\, dt + \boldsymbol{\sigma}(\boldsymbol{\theta}) d\boldsymbol{\omega}_t \tag{A.5}$$

$$\boldsymbol{y}_k = \boldsymbol{C}(\boldsymbol{\theta})\boldsymbol{x}_k + \boldsymbol{D}(\boldsymbol{\theta})\boldsymbol{u}_k + \boldsymbol{e}_k \tag{A.6}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a state vector, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is an input vector, $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is an output vector, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{A}(\cdot) \in \mathbb{R}^{n\times n}$, $\boldsymbol{B}(\cdot) \in \mathbb{R}^{n\times m}$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n\times n}$, $\boldsymbol{C}(\cdot) \in \mathbb{R}^{l\times n}$ and $\boldsymbol{D}(\cdot) \in \mathbb{R}^{l\times m}$ are nonlinear functions, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{\theta}))$.

A.1.2 Parameter estimation methods

CTSM allows a number of different methods to be applied to estimate the parameters of the above model structures as outlined in the following.

A.1.2.1 Maximum likelihood estimation

Given a particular model structure, *maximum likelihood* (ML) estimation of the unknown parameters can be performed by finding the parameters $\boldsymbol{\theta}$ that maximize the likelihood function of a given sequence of measurements $\boldsymbol{y}_0$, $\boldsymbol{y}_1$, $\ldots$, $\boldsymbol{y}_k$, $\ldots$, $\boldsymbol{y}_N$. By introducing the notation:

$$\mathcal{Y}_k = [\boldsymbol{y}_k, \boldsymbol{y}_{k-1}, \ldots, \boldsymbol{y}_1, \boldsymbol{y}_0] \tag{A.7}$$

the likelihood function is the joint probability density:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = p(\mathcal{Y}_N|\boldsymbol{\theta}) \tag{A.8}$$

or equivalently:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} p(\boldsymbol{y}_k | \mathcal{Y}_{k-1}, \boldsymbol{\theta}) \right) p(\boldsymbol{y}_0 | \boldsymbol{\theta}) \tag{A.9}$$

where the rule $P(A \cap B) = P(A|B)P(B)$ has been applied to form a product of conditional probability densities. In order to obtain an exact evaluation of the likelihood function, the initial probability density $p(\boldsymbol{y}_0|\boldsymbol{\theta})$ must be known and all subsequent conditional densities must be determined by successively solving Kolmogorov's forward equation and applying Bayes' rule (Jazwinski, 1970), but this approach is computationally infeasible in practice. However, since the diffusion terms in the above model structures do not depend on the state variables, a simpler alternative can be used. More specifically, a method based on Kalman filtering can be applied for LTI and LTV models, and an approximate method based on extended Kalman filtering can be applied for NL models. The latter approximation can be applied, because the stochastic differential equations considered are driven by Wiener processes, and because increments of a Wiener process are Gaussian, which makes it reasonable to assume, under some regularity conditions, that the conditional densities can be well approximated by Gaussian densities. The Gaussian density is completely characterized by its mean and covariance, so by introducing the notation:

$$\hat{\boldsymbol{y}}_{k|k-1} = E\{\boldsymbol{y}_k | \mathcal{Y}_{k-1}, \boldsymbol{\theta}\} \tag{A.10}$$

$$\boldsymbol{R}_{k|k-1} = V\{\boldsymbol{y}_k | \mathcal{Y}_{k-1}, \boldsymbol{\theta}\} \tag{A.11}$$

and:

$$\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{A.12}$$

the likelihood function can be written as follows:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1} \boldsymbol{\epsilon}_k\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1})}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0 | \boldsymbol{\theta}) \tag{A.13}$$

where, for given parameters and initial states, $\boldsymbol{\epsilon}_k$ and $\boldsymbol{R}_{k|k-1}$ can be computed by means of a Kalman filter (LTI and LTV models) or an extended Kalman filter (NL models) as shown in Sections A.1.3.1 and A.1.3.2 respectively. Further conditioning on $\boldsymbol{y}_0$ and taking the negative logarithm gives:

$$\begin{aligned} -\ln\left(L(\boldsymbol{\theta}; \mathcal{Y}_N | \boldsymbol{y}_0)\right) = \frac{1}{2} \sum_{k=1}^{N} \left(\ln(\det(\boldsymbol{R}_{k|k-1})) + \boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1} \boldsymbol{\epsilon}_k \right) \\ + \frac{1}{2} \left(\sum_{k=1}^{N} l \right) \ln(2\pi) \end{aligned} \tag{A.14}$$

and ML estimates of the parameters (and optionally of the initial states) can now be determined by solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \Theta} \left\{ -\ln\left(L(\boldsymbol{\theta}; \mathcal{Y}_N | \boldsymbol{y}_0)\right) \right\} \tag{A.15}$$

A.1.2.2 Maximum a posteriori estimation

If prior information about the parameters is available in the form of a prior probability density function $p(\boldsymbol{\theta})$, Bayes' rule can be applied to give an improved estimate by forming the posterior probability density function:

$$p(\boldsymbol{\theta}|\mathcal{Y}_N) = \frac{p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{Y}_N)} \propto p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{A.16}$$

and subsequently finding the parameters that maximize this function, i.e. by performing *maximum a posteriori* (MAP) estimation. A nice feature of this expression is the fact that it reduces to the likelihood function, when no prior information is available ($p(\boldsymbol{\theta})$ uniform), making ML estimation a special case of MAP estimation. In fact, this formulation also allows MAP estimation on a subset of the parameters ($p(\boldsymbol{\theta})$ partly uniform). By introducing the notation[1]:

$$\boldsymbol{\mu}_{\boldsymbol{\theta}} = E\{\boldsymbol{\theta}\} \tag{A.17}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{\theta}} = V\{\boldsymbol{\theta}\} \tag{A.18}$$

and:

$$\boldsymbol{\epsilon}_{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}} \tag{A.19}$$

and by assuming that the prior probability density of the parameters is Gaussian, the posterior probability density function can be written as follows:

$$\begin{aligned} p(\boldsymbol{\theta}|\mathcal{Y}_N) \propto & \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1} \boldsymbol{\epsilon}_k\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1})}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \\ & \times \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det(\boldsymbol{\Sigma}_{\boldsymbol{\theta}})}\left(\sqrt{2\pi}\right)^p} \end{aligned} \tag{A.20}$$

Further conditioning on $\boldsymbol{y}_0$ and taking the negative logarithm gives:

$$\begin{aligned} -\ln\left(p(\boldsymbol{\theta}|\mathcal{Y}_N, \boldsymbol{y}_0)\right) \propto & \frac{1}{2}\sum_{k=1}^{N}\left(\ln(\det(\boldsymbol{R}_{k|k-1})) + \boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1}\boldsymbol{\epsilon}_k\right) \\ & + \frac{1}{2}\left(\left(\sum_{k=1}^{N} l\right) + p\right)\ln(2\pi) \\ & + \frac{1}{2}\ln(\det(\boldsymbol{\Sigma}_{\boldsymbol{\theta}})) + \frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{\epsilon}_{\boldsymbol{\theta}} \end{aligned} \tag{A.21}$$

and MAP estimates of the parameters (and optionally of the initial states) can now be determined by solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta}\left\{-\ln\left(p(\boldsymbol{\theta}|\mathcal{Y}_N, \boldsymbol{y}_0)\right)\right\} \tag{A.22}$$

[1] In practice $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ is specified as $\boldsymbol{\Sigma}_{\boldsymbol{\theta}} = \boldsymbol{\sigma}_{\boldsymbol{\theta}}\boldsymbol{R}_{\boldsymbol{\theta}}\boldsymbol{\sigma}_{\boldsymbol{\theta}}$, where $\boldsymbol{\sigma}_{\boldsymbol{\theta}}$ is a diagonal matrix of the prior standard deviations and $\boldsymbol{R}_{\boldsymbol{\theta}}$ is the corresponding prior correlation matrix.

A.1.2.3 Using multiple independent data sets

If, instead of a single sequence of measurements, multiple consecutive, but yet separate, sequences of measurements, i.e. $\mathcal{Y}_{N_1}^1$, $\mathcal{Y}_{N_2}^2$, ..., $\mathcal{Y}_{N_i}^i$, ..., $\mathcal{Y}_{N_S}^S$, are available, a similar estimation method can be applied by expanding the expression for the posterior probability density function to the general form:

$$p(\boldsymbol{\theta}|\mathbf{Y}) \propto \left(\prod_{i=1}^{S} \left(\prod_{k=1}^{N_i} \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1}^i)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0^i|\boldsymbol{\theta}) \right) \times \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det(\boldsymbol{\Sigma}_{\boldsymbol{\theta}})}\left(\sqrt{2\pi}\right)^p} \tag{A.23}$$

where:

$$\mathbf{Y} = [\mathcal{Y}_{N_1}^1, \mathcal{Y}_{N_2}^2, \ldots, \mathcal{Y}_{N_i}^i, \ldots, \mathcal{Y}_{N_S}^S] \tag{A.24}$$

and where the individual sequences of measurements are assumed to be stochastically independent. This formulation allows MAP estimation on multiple data sets, but, as special cases, it also allows ML estimation on multiple data sets ($p(\boldsymbol{\theta})$ uniform), MAP estimation on a single data set ($S = 1$) and ML estimation on a single data set ($p(\boldsymbol{\theta})$ uniform, $S = 1$). Further conditioning on:

$$\mathbf{y_0} = [\boldsymbol{y}_0^1, \boldsymbol{y}_0^2, \ldots, \boldsymbol{y}_0^i, \ldots, \boldsymbol{y}_0^S] \tag{A.25}$$

and taking the negative logarithm gives:

$$\begin{aligned} -\ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right) \propto & \frac{1}{2} \sum_{i=1}^{S} \sum_{k=1}^{N_i} \left(\ln(\det(\boldsymbol{R}_{k|k-1}^i)) + (\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i \right) \\ & + \frac{1}{2} \left(\left(\sum_{i=1}^{S} \sum_{k=1}^{N_i} l \right) + p \right) \ln(2\pi) \\ & + \frac{1}{2} \ln(\det(\boldsymbol{\Sigma}_{\boldsymbol{\theta}})) + \frac{1}{2} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\theta}} \end{aligned} \tag{A.26}$$

and estimates of the parameters (and optionally of the initial states) can now be determined by solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \Theta} \left\{ -\ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right) \right\} \tag{A.27}$$

A.1.3 Filtering methods

CTSM computes the innovation vectors $\boldsymbol{\epsilon}_k$ (or $\boldsymbol{\epsilon}_k^i$) and their covariance matrices $\boldsymbol{R}_{k|k-1}$ (or $\boldsymbol{R}_{k|k-1}^i$) recursively by means of a Kalman filter (LTI and LTV models) or an extended Kalman filter (NL models) as outlined in the following.

A.1.3.1 Kalman filtering

For LTI and LTV models $\boldsymbol{\epsilon}_k$ (or $\boldsymbol{\epsilon}_k^i$) and $\boldsymbol{R}_{k|k-1}$ (or $\boldsymbol{R}_{k|k-1}^i$) can be computed for a given set of parameters $\boldsymbol{\theta}$ and initial states $\boldsymbol{x}_0$ by means of a continuous-discrete Kalman filter, i.e. by means of the output *prediction* equations:

$$\hat{\boldsymbol{y}}_{k|k-1} = \boldsymbol{C}\hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{D}\boldsymbol{u}_k \tag{A.28}$$

$$\boldsymbol{R}_{k|k-1} = \boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^T + \boldsymbol{S} \tag{A.29}$$

the *innovation* equation:

$$\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{A.30}$$

the Kalman *gain* equation:

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{R}_{k|k-1}^{-1} \tag{A.31}$$

the *updating* equations:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\boldsymbol{\epsilon}_k \tag{A.32}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{R}_{k|k-1}\boldsymbol{K}_k^T \tag{A.33}$$

and the state *prediction* equations:

$$\frac{d\hat{\boldsymbol{x}}_{t|k}}{dt} = \boldsymbol{A}\hat{\boldsymbol{x}}_{t|k} + \boldsymbol{B}\boldsymbol{u}_t \ , \ t \in [t_k, t_{k+1}[\tag{A.34}$$

$$\frac{d\boldsymbol{P}_{t|k}}{dt} = \boldsymbol{A}\boldsymbol{P}_{t|k} + \boldsymbol{P}_{t|k}\boldsymbol{A}^T + \boldsymbol{\sigma}\boldsymbol{\sigma}^T \ , \ t \in [t_k, t_{k+1}[\tag{A.35}$$

where the following shorthand notation applies in the LTV case:

$$\begin{gathered}\boldsymbol{A} = \boldsymbol{A}(\hat{\boldsymbol{x}}_{t|k-1}, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \ , \ \boldsymbol{B} = \boldsymbol{B}(\hat{\boldsymbol{x}}_{t|k-1}, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \\ \boldsymbol{C} = \boldsymbol{C}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \ , \ \boldsymbol{D} = \boldsymbol{D}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta}) \ , \ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta})\end{gathered} \tag{A.36}$$

and the following shorthand notation applies in the LTI case:

$$\begin{gathered}\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{\theta}) \ , \ \boldsymbol{B} = \boldsymbol{B}(\boldsymbol{\theta}) \\ \boldsymbol{C} = \boldsymbol{C}(\boldsymbol{\theta}) \ , \ \boldsymbol{D} = \boldsymbol{D}(\boldsymbol{\theta}) \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\theta}) \ , \ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{\theta})\end{gathered} \tag{A.37}$$

Initial conditions for the Kalman filter are $\hat{\boldsymbol{x}}_{t|t_0} = \boldsymbol{x}_0$ and $\boldsymbol{P}_{t|t_0} = \boldsymbol{P}_0$, which may either be pre-specified or estimated along with the parameters as a part of the overall problem (see Section A.1.3.4). In the LTI case, and in the LTV case, if $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{\sigma}$ and $\boldsymbol{S}$ are assumed constant between samples[2], (A.34)

[2]In practice the time interval $t \in [t_k, t_{k+1}[$ is subsampled for LTV models, and $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{\sigma}$ and $\boldsymbol{S}$ are evaluated at each subsampling instant to provide a better approximation.

and (A.35) can be replaced by their discrete time counterparts, which can be derived from the solution to the stochastic differential equation:

$$d\boldsymbol{x}_t = (\boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}\boldsymbol{u}_t)\,dt + \boldsymbol{\sigma} d\boldsymbol{\omega}_t \ , \ t \in [t_k, t_{k+1}[\tag{A.38}$$

i.e. from:

$$\boldsymbol{x}_{t_{k+1}} = e^{\boldsymbol{A}(t_{k+1}-t_k)}\boldsymbol{x}_{t_k} + \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{B}\boldsymbol{u}_s ds + \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{\sigma} d\boldsymbol{\omega}_s \tag{A.39}$$

which yields:

$$\hat{\boldsymbol{x}}_{k+1|k} = E\{\boldsymbol{x}_{t_{k+1}}|\boldsymbol{x}_{t_k}\} = e^{\boldsymbol{A}(t_{k+1}-t_k)}\hat{\boldsymbol{x}}_{k|k} + \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{B}\boldsymbol{u}_s ds \tag{A.40}$$

$$\begin{aligned}\boldsymbol{P}_{k+1|k} = E\{\boldsymbol{x}_{t_{k+1}}\boldsymbol{x}_{t_{k+1}}^T|\boldsymbol{x}_{t_k}\} &= e^{\boldsymbol{A}(t_{k+1}-t_k)}\boldsymbol{P}_{k|k}\left(e^{\boldsymbol{A}(t_{k+1}-t_k)}\right)^T \\ &+ \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}(t_{k+1}-s)}\right)^T ds\end{aligned} \tag{A.41}$$

where the following shorthand notation applies in the LTV case:

$$\begin{aligned}\boldsymbol{A} = \boldsymbol{A}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \ , \ \boldsymbol{B} = \boldsymbol{B}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \\ \boldsymbol{C} = \boldsymbol{C}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \ , \ \boldsymbol{D} = \boldsymbol{D}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \ , \ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta})\end{aligned} \tag{A.42}$$

and the following shorthand notation applies in the LTI case:

$$\begin{aligned}\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{\theta}) \ , \ \boldsymbol{B} = \boldsymbol{B}(\boldsymbol{\theta}) \\ \boldsymbol{C} = \boldsymbol{C}(\boldsymbol{\theta}) \ , \ \boldsymbol{D} = \boldsymbol{D}(\boldsymbol{\theta}) \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\theta}) \ , \ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{\theta})\end{aligned} \tag{A.43}$$

In order to be able to use (A.40) and (A.41), the integrals of both equations must be computed. For this purpose the equations are rewritten to:

$$\begin{aligned}\hat{\boldsymbol{x}}_{k+1|k} &= e^{\boldsymbol{A}(t_{k+1}-t_k)}\hat{\boldsymbol{x}}_{k|k} + \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{B}\boldsymbol{u}_s ds \\ &= e^{\boldsymbol{A}\tau_s}\hat{\boldsymbol{x}}_{k|k} + \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{B}\left(\boldsymbol{\alpha}(s-t_k) + \boldsymbol{u}_k\right) ds \\ &= \boldsymbol{\Phi}_s\hat{\boldsymbol{x}}_{k|k} + \int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{B}\left(\boldsymbol{\alpha}(\tau_s - s) + \boldsymbol{u}_k\right) ds \\ &= \boldsymbol{\Phi}_s\hat{\boldsymbol{x}}_{k|k} - \int_0^{\tau_s} e^{\boldsymbol{A}s} s ds \boldsymbol{B}\boldsymbol{\alpha} + \int_0^{\tau_s} e^{\boldsymbol{A}s} ds \boldsymbol{B}\left(\boldsymbol{\alpha}\tau_s + \boldsymbol{u}_k\right)\end{aligned} \tag{A.44}$$

and:

$$\begin{aligned}\boldsymbol{P}_{k+1|k} &= e^{\boldsymbol{A}(t_{k+1}-t_k)}\boldsymbol{P}_{k|k}\left(e^{\boldsymbol{A}(t_{k+1}-t_k)}\right)^T \\ &\quad + \int_{t_k}^{t_{k+1}} e^{\boldsymbol{A}(t_{k+1}-s)}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}(t_{k+1}-s)}\right)^T ds \\ &= e^{\boldsymbol{A}\tau_s}\boldsymbol{P}_{k|k}\left(e^{\boldsymbol{A}\tau_s}\right)^T + \int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}s}\right)^T ds \\ &= \boldsymbol{\Phi}_s\boldsymbol{P}_{k|k}\boldsymbol{\Phi}_s^T + \int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}s}\right)^T ds\end{aligned} \tag{A.45}$$

where $\tau_s = t_{k+1} - t_k$ and $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$, and where:

$$\boldsymbol{\alpha} = \frac{\boldsymbol{u}_{k+1} - \boldsymbol{u}_k}{t_{k+1} - t_k} \tag{A.46}$$

has been introduced to allow assumption of either *zero order hold* ($\boldsymbol{\alpha} = \boldsymbol{0}$) or *first order hold* ($\boldsymbol{\alpha} \neq \boldsymbol{0}$) on the inputs between sampling instants. The matrix exponential $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$ can be computed by means of a Padé approximation with repeated scaling and squaring (Moler and van Loan, 1978). However, both $\boldsymbol{\Phi}_s$ and the integral in (A.45) can be computed simultaneously through:

$$\exp\left(\begin{bmatrix}-\boldsymbol{A} & \boldsymbol{\sigma}\boldsymbol{\sigma}^T \\ \boldsymbol{0} & \boldsymbol{A}^T\end{bmatrix}\tau_s\right) = \begin{bmatrix}\boldsymbol{H}_1(\tau_s) & \boldsymbol{H}_2(\tau_s) \\ \boldsymbol{0} & \boldsymbol{H}_3(\tau_s)\end{bmatrix} \tag{A.47}$$

by combining submatrices of the result[3] (van Loan, 1978), i.e.:

$$\boldsymbol{\Phi}_s = \boldsymbol{H}_3^T(\tau_s) \tag{A.48}$$

and:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}s}\right)^T ds = \boldsymbol{H}_3^T(\tau_s)\boldsymbol{H}_2(\tau_s) \tag{A.49}$$

Alternatively, this integral can be computed from the Lyapunov equation:

$$\begin{aligned}\boldsymbol{\Phi}_s\boldsymbol{\sigma}\boldsymbol{\sigma}^T\boldsymbol{\Phi}_s^T - \boldsymbol{\sigma}\boldsymbol{\sigma}^T &= \boldsymbol{A}\int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}s}\right)^T ds \\ &\quad + \int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T\left(e^{\boldsymbol{A}s}\right)^T ds\boldsymbol{A}^T\end{aligned} \tag{A.50}$$

but this approach has been found to be less feasible. The integrals in (A.44) are not as easy to deal with, especially if $\boldsymbol{A}$ is singular. However, this problem can be solved by introducing the singular value decomposition (SVD) of $\boldsymbol{A}$, i.e. $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, transforming the integrals and subsequently computing these.

[3] Within **CTSM** the specific implementation is based on the algorithms of Sidje (1998).

The first integral can be transformed as follows:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} s ds = \boldsymbol{U} \int_0^{\tau_s} \boldsymbol{U}^T e^{\boldsymbol{A}s} \boldsymbol{U} s ds \boldsymbol{U}^T = \boldsymbol{U} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} s ds \boldsymbol{U}^T \tag{A.51}$$

and, if $\boldsymbol{A}$ is singular, the matrix $\tilde{\boldsymbol{A}} = \boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{U} = \boldsymbol{U}^T\boldsymbol{A}\boldsymbol{U}$ has a special structure:

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \tag{A.52}$$

which allows the integral to be computed as follows:

$$\begin{aligned}
\int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} s ds &= \int_0^{\tau_s} \left(\boldsymbol{I}s + \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} s^2 + \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}^2 \frac{s^3}{2} + \cdots \right) ds \\
&= \int_0^{\tau_s} \left(\boldsymbol{I}s + \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} s^2 + \begin{bmatrix} \tilde{\boldsymbol{A}}_1^2 & \tilde{\boldsymbol{A}}_1\tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \frac{s^3}{2} + \cdots \right) ds \\
&= \begin{bmatrix} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}_1 s} s ds & \int_0^{\tau_s} \tilde{\boldsymbol{A}}_1^{-1} \left(e^{\tilde{\boldsymbol{A}}_1 s} - \boldsymbol{I} \right) s \tilde{\boldsymbol{A}}_2 ds \\ \boldsymbol{0} & \boldsymbol{I}\frac{\tau_s^2}{2} \end{bmatrix} \\
&= \begin{bmatrix} \left[\tilde{\boldsymbol{A}}_1^{-1} e^{\tilde{\boldsymbol{A}}_1 s} \left(\boldsymbol{I}s - \tilde{\boldsymbol{A}}_1^{-1} \right) \right]_0^{\tau_s} & \\ \boldsymbol{0} & \\ & \tilde{\boldsymbol{A}}_1^{-1} \left[\tilde{\boldsymbol{A}}_1^{-1} e^{\tilde{\boldsymbol{A}}_1 s} \left(\boldsymbol{I}s - \tilde{\boldsymbol{A}}_1^{-1} \right) - \boldsymbol{I}\frac{s^2}{2} \right]_0^{\tau_s} \tilde{\boldsymbol{A}}_2 \\ & \boldsymbol{I}\frac{\tau_s^2}{2} \end{bmatrix} \\
&= \begin{bmatrix} \tilde{\boldsymbol{A}}_1^{-1} \left(-\tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) + \tilde{\boldsymbol{\Phi}}_s^1 \tau_s \right) & \\ \boldsymbol{0} & \\ & \tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{A}}_1^{-1} \left(-\tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) + \tilde{\boldsymbol{\Phi}}_s^1 \tau_s \right) - \boldsymbol{I}\frac{\tau_s^2}{2} \right) \tilde{\boldsymbol{A}}_2 \\ & \boldsymbol{I}\frac{\tau_s^2}{2} \end{bmatrix}
\end{aligned} \tag{A.53}$$

where $\tilde{\boldsymbol{\Phi}}_s^1$ is the upper left part of the matrix:

$$\tilde{\boldsymbol{\Phi}}_s = \boldsymbol{U}^T \boldsymbol{\Phi}_s \boldsymbol{U} = \begin{bmatrix} \tilde{\boldsymbol{\Phi}}_s^1 & \tilde{\boldsymbol{\Phi}}_s^2 \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \tag{A.54}$$

The second integral can be transformed as follows:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} ds = \boldsymbol{U} \int_0^{\tau_s} \boldsymbol{U}^T e^{\boldsymbol{A}s} \boldsymbol{U} ds \boldsymbol{U}^T = \boldsymbol{U} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds \boldsymbol{U}^T \tag{A.55}$$

and can subsequently be computed as follows:

$$\begin{aligned}
\int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds &= \int_0^{\tau_s} \left(\boldsymbol{I} + \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} s + \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}^2 \frac{s^2}{2} + \cdots \right) ds \\
&= \int_0^{\tau_s} \left(\boldsymbol{I} + \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} s + \begin{bmatrix} \tilde{\boldsymbol{A}}_1^2 & \tilde{\boldsymbol{A}}_1\tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \frac{s^2}{2} + \cdots \right) ds \\
&= \begin{bmatrix} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}_1 s} ds & \int_0^{\tau_s} \tilde{\boldsymbol{A}}_1^{-1} \left(e^{\tilde{\boldsymbol{A}}_1 s} - \boldsymbol{I} \right) \tilde{\boldsymbol{A}}_2 ds \\ \boldsymbol{0} & \boldsymbol{I}\tau_s \end{bmatrix} \\
&= \begin{bmatrix} \left[\tilde{\boldsymbol{A}}_1^{-1} e^{\tilde{\boldsymbol{A}}_1 s} \right]_0^{\tau_s} & \tilde{\boldsymbol{A}}_1^{-1} \left[\tilde{\boldsymbol{A}}_1^{-1} e^{\tilde{\boldsymbol{A}}_1 s} - \boldsymbol{I}s \right]_0^{\tau_s} \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{I}\tau_s \end{bmatrix} \\
&= \begin{bmatrix} \tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) & \tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) - \boldsymbol{I}\tau_s \right) \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{I}\tau_s \end{bmatrix}
\end{aligned} \tag{A.56}$$

Depending on the specific singularity of $\boldsymbol{A}$ (see Section A.1.3.3 for details on how this is determined in **CTSM**) and the particular nature of the inputs, several different cases are possible as shown in the following.

General case: Singular $\boldsymbol{A}$, first order hold on inputs

In the general case, the Kalman filter prediction can be calculated as follows:

$$\hat{\boldsymbol{x}}_{j+1} = \boldsymbol{\Phi}_s \hat{\boldsymbol{x}}_j - \boldsymbol{U} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds \boldsymbol{U}^T \boldsymbol{B}\boldsymbol{\alpha} + \boldsymbol{U} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds \boldsymbol{U}^T \boldsymbol{B} \left(\boldsymbol{\alpha}\tau_s + \boldsymbol{u}_j \right) \tag{A.57}$$

with:

$$\int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds = \begin{bmatrix} \tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) & \tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) - \boldsymbol{I}\tau_s \right) \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{I}\tau_s \end{bmatrix} \tag{A.58}$$

and:

$$\begin{aligned}
\int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} s ds = \begin{bmatrix} \tilde{\boldsymbol{A}}_1^{-1} \left(-\tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) + \tilde{\boldsymbol{\Phi}}_s^1 \tau_s \right) \\ \boldsymbol{0} \end{bmatrix} \\
\begin{bmatrix} \tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{A}}_1^{-1} \left(-\tilde{\boldsymbol{A}}_1^{-1} \left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I} \right) + \tilde{\boldsymbol{\Phi}}_s^1 \tau_s \right) - \boldsymbol{I}\frac{\tau_s^2}{2} \right) \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{I}\frac{\tau_s^2}{2} \end{bmatrix}
\end{aligned} \tag{A.59}$$

Special case no. 1: Singular $\boldsymbol{A}$, zero order hold on inputs

The Kalman filter prediction for this special case can be calculated as follows:

$$\hat{\boldsymbol{x}}_{j+1} = \boldsymbol{\Phi}_s \hat{\boldsymbol{x}}_j + \boldsymbol{U} \int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds \boldsymbol{U}^T \boldsymbol{B} \boldsymbol{u}_j \tag{A.60}$$

with:

$$\int_0^{\tau_s} e^{\tilde{\boldsymbol{A}}s} ds = \begin{bmatrix} \tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I}\right) & \tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I}\right) - \boldsymbol{I}\tau_s\right)\tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{I}\tau_s \end{bmatrix} \tag{A.61}$$

Special case no. 2: Nonsingular $\boldsymbol{A}$, first order hold on inputs

The Kalman filter prediction for this special case can be calculated as follows:

$$\hat{\boldsymbol{x}}_{j+1} = \boldsymbol{\Phi}_s \hat{\boldsymbol{x}}_j - \int_0^{\tau_s} e^{\boldsymbol{A}s} s ds \boldsymbol{B}\boldsymbol{\alpha} + \int_0^{\tau_s} e^{\boldsymbol{A}s} ds \boldsymbol{B}\left(\boldsymbol{\alpha}\tau_s + \boldsymbol{u}_j\right) \tag{A.62}$$

with:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} ds = \boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right) \tag{A.63}$$

and:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} s ds = \boldsymbol{A}^{-1}\left(-\boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right) + \boldsymbol{\Phi}_s\tau_s\right) \tag{A.64}$$

Special case no. 3: Nonsingular $\boldsymbol{A}$, zero order hold on inputs

The Kalman filter prediction for this special case can be calculated as follows:

$$\hat{\boldsymbol{x}}_{j+1} = \boldsymbol{\Phi}_s \hat{\boldsymbol{x}}_j + \int_0^{\tau_s} e^{\boldsymbol{A}s} ds \boldsymbol{B}\boldsymbol{u}_j \tag{A.65}$$

with:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} ds = \boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right) \tag{A.66}$$

Special case no. 4: Identically zero $\boldsymbol{A}$, first order hold on inputs

The Kalman filter prediction for this special case can be calculated as follows:

$$\hat{\boldsymbol{x}}_{j+1} = \hat{\boldsymbol{x}}_j - \int_0^{\tau_s} e^{\boldsymbol{A}s} s ds \boldsymbol{B}\boldsymbol{\alpha} + \int_0^{\tau_s} e^{\boldsymbol{A}s} ds \boldsymbol{B}\left(\boldsymbol{\alpha}\tau_s + \boldsymbol{u}_j\right) \tag{A.67}$$

with:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} ds = \boldsymbol{I}\tau_s \tag{A.68}$$

and:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} s ds = \boldsymbol{I}\frac{\tau_s^2}{2} \tag{A.69}$$

Special case no. 5: Identically zero $\boldsymbol{A}$, zero order hold on inputs

The Kalman filter prediction for this special case can be calculated as follows:

$$\hat{\boldsymbol{x}}_{j+1} = \hat{\boldsymbol{x}}_j + \int_0^{\tau_s} e^{\boldsymbol{A}s} ds \boldsymbol{B}\boldsymbol{u}_j \tag{A.70}$$

with:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s} ds = \boldsymbol{I}\tau_s \tag{A.71}$$

A.1.3.2 Extended Kalman filtering

For NL models $\boldsymbol{\epsilon}_k$ (or $\boldsymbol{\epsilon}_k^i$) and $\boldsymbol{R}_{k|k-1}$ (or $\boldsymbol{R}_{k|k-1}^i$) can be computed for a given set of parameters $\boldsymbol{\theta}$ and initial states $\boldsymbol{x}_0$ by means of a continuous-discrete extended Kalman filter, i.e. by means of the output *prediction* equations:

$$\hat{\boldsymbol{y}}_{k|k-1} = \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \tag{A.72}$$

$$\boldsymbol{R}_{k|k-1} = \boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^T + \boldsymbol{S} \tag{A.73}$$

the *innovation* equation:

$$\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{A.74}$$

the Kalman *gain* equation:

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{R}_{k|k-1}^{-1} \tag{A.75}$$

the *updating* equations:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\boldsymbol{\epsilon}_k \tag{A.76}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{R}_{k|k-1}\boldsymbol{K}_k^T \tag{A.77}$$

and the state *prediction* equations:

$$\frac{d\hat{\boldsymbol{x}}_{t|k}}{dt} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{t|k}, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \ , \ t \in [t_k, t_{k+1}[\tag{A.78}$$

$$\frac{d\boldsymbol{P}_{t|k}}{dt} = \boldsymbol{A}\boldsymbol{P}_{t|k} + \boldsymbol{P}_{t|k}\boldsymbol{A}^T + \boldsymbol{\sigma}\boldsymbol{\sigma}^T \ , \ t \in [t_k, t_{k+1}[\tag{A.79}$$

where the following shorthand notation has been applied[4]:

$$\boldsymbol{A} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_t}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}=\boldsymbol{u}_k, t=t_k, \boldsymbol{\theta}} , \ \boldsymbol{C} = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_t}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}=\boldsymbol{u}_k, t=t_k, \boldsymbol{\theta}} \tag{A.80}$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \ , \ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta})$$

[4]Within **CTSM** the code needed to evaluate the Jacobians is generated through analytical manipulation using a method based on the algorithms of Speelpenning (1980).

Initial conditions for the extended Kalman filter are $\hat{\boldsymbol{x}}_{t|t_0} = \boldsymbol{x}_0$ and $\boldsymbol{P}_{t|t_0} = \boldsymbol{P}_0$, which may either be pre-specified or estimated along with the parameters as a part of the overall problem (see Section A.1.3.4). Being a linear filter, the extended Kalman filter is sensitive to nonlinear effects, and the approximate solution obtained by solving (A.78) and (A.79) may be too crude (Jazwinski, 1970). Moreover, the assumption of Gaussian conditional densities is only likely to hold for small sample times. To provide a better approximation, the time interval $[t_k, t_{k+1}[$ is therefore subsampled, i.e. $[t_k, \ldots, t_j, \ldots, t_{k+1}[$, and the equations are linearized at each subsampling instant. This also means that direct numerical solution of (A.78) and (A.79) can be avoided by applying the analytical solutions to the corresponding linearized propagation equations:

$$\frac{d\hat{\boldsymbol{x}}_{t|j}}{dt} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{j|j-1}, \boldsymbol{u}_j, t_j, \boldsymbol{\theta}) + \boldsymbol{A}(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{B}(\boldsymbol{u}_t - \boldsymbol{u}_j),\ t \in [t_j, t_{j+1}[\tag{A.81}$$

$$\frac{d\boldsymbol{P}_{t|j}}{dt} = \boldsymbol{A}\boldsymbol{P}_{t|j} + \boldsymbol{P}_{t|j}\boldsymbol{A}^T + \boldsymbol{\sigma}\boldsymbol{\sigma}^T,\ t \in [t_j, t_{j+1}[\tag{A.82}$$

where the following shorthand notation has been applied[5]:

$$\boldsymbol{A} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_t}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{j|j-1}, \boldsymbol{u}=\boldsymbol{u}_j, t=t_j, \boldsymbol{\theta}},\ \boldsymbol{B} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}_t}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{j|j-1}, \boldsymbol{u}=\boldsymbol{u}_j, t=t_j, \boldsymbol{\theta}} \tag{A.83}$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_j, t_j, \boldsymbol{\theta})\ ,\ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{u}_j, t_j, \boldsymbol{\theta})$$

The solution to (A.82) is equivalent to the solution to (A.35), i.e.:

$$\boldsymbol{P}_{j+1|j} = \boldsymbol{\Phi}_s \boldsymbol{P}_{j|j} \boldsymbol{\Phi}_s^T + \int_0^{\tau_s} e^{\boldsymbol{A}s} \boldsymbol{\sigma}\boldsymbol{\sigma}^T \left(e^{\boldsymbol{A}s}\right)^T ds \tag{A.84}$$

where $\tau_s = t_{j+1} - t_j$ and $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$. The solution to (A.81) is not as easy to find, especially if $\boldsymbol{A}$ is singular. Nevertheless, by simplifying the notation, i.e.:

$$\frac{d\hat{\boldsymbol{x}}_t}{dt} = \boldsymbol{f} + \boldsymbol{A}(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{B}(\boldsymbol{u}_t - \boldsymbol{u}_j)\ ,\ t \in [t_j, t_{j+1}[\tag{A.85}$$

and introducing:

$$\boldsymbol{\alpha} = \frac{\boldsymbol{u}_{j+1} - \boldsymbol{u}_j}{t_{j+1} - t_j} \tag{A.86}$$

to allow assumption of either *zero order hold* ($\boldsymbol{\alpha} = \boldsymbol{0}$) or *first order hold* ($\boldsymbol{\alpha} \neq \boldsymbol{0}$) on the inputs between sampling instants, i.e.:

$$\frac{d\hat{\boldsymbol{x}}_t}{dt} = \boldsymbol{f} + \boldsymbol{A}(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{B}(\boldsymbol{\alpha}(t - t_j) + \boldsymbol{u}_j - \boldsymbol{u}_j)\ ,\ t \in [t_j, t_{j+1}[\tag{A.87}$$

[5] Within **CTSM** the code needed to evaluate the Jacobians is generated through analytical manipulation using a method based on the algorithms of Speelpenning (1980).

and by introducing the singular value decomposition (SVD) of $\boldsymbol{A}$, i.e. $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, a solvable equation can be obtained as follows:

$$\begin{aligned}
\frac{d\hat{\boldsymbol{x}}_t}{dt} &= \boldsymbol{f} + \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{B}\boldsymbol{\alpha}(t - t_j) \\
\boldsymbol{U}^T\frac{d\hat{\boldsymbol{x}}_t}{dt} &= \boldsymbol{U}^T\boldsymbol{f} + \boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{U}\boldsymbol{U}^T(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{U}^T\boldsymbol{B}\boldsymbol{\alpha}(t - t_j) \\
\frac{d\boldsymbol{z}_t}{dt} &= \boldsymbol{U}^T\boldsymbol{f} + \boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{U}(\boldsymbol{z}_t - \boldsymbol{z}_j) + \boldsymbol{U}^T\boldsymbol{B}\boldsymbol{\alpha}(t - t_j) \\
\frac{d\boldsymbol{z}_t}{dt} &= \tilde{\boldsymbol{f}} + \tilde{\boldsymbol{A}}(\boldsymbol{z}_t - \boldsymbol{z}_j) + \tilde{\boldsymbol{B}}\boldsymbol{\alpha}(t - t_j)\ ,\ t \in [t_j, t_{j+1}[
\end{aligned} \tag{A.88}$$

where the transformation $\boldsymbol{z}_t = \boldsymbol{U}^T\hat{\boldsymbol{x}}_t$ has been introduced along with the vector $\tilde{\boldsymbol{f}} = \boldsymbol{U}^T\boldsymbol{f}$ and the matrices $\tilde{\boldsymbol{A}} = \boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{U} = \boldsymbol{U}^T\boldsymbol{A}\boldsymbol{U}$ and $\tilde{\boldsymbol{B}} = \boldsymbol{U}^T\boldsymbol{B}$. Now, if $\boldsymbol{A}$ is singular, the matrix $\tilde{\boldsymbol{A}}$ has a special structure:

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} \tilde{\boldsymbol{A}}_1 & \tilde{\boldsymbol{A}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \tag{A.89}$$

which makes it possible to split up the previous result in two distinct equations:

$$\begin{aligned}
\frac{d\boldsymbol{z}_t^1}{dt} &= \tilde{\boldsymbol{f}}_1 + \tilde{\boldsymbol{A}}_1(\boldsymbol{z}_t^1 - \boldsymbol{z}_j^1) + \tilde{\boldsymbol{A}}_2(\boldsymbol{z}_t^2 - \boldsymbol{z}_j^2) + \tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}(t - t_j),\ t \in [t_j, t_{j+1}[\\
\frac{d\boldsymbol{z}_t^2}{dt} &= \tilde{\boldsymbol{f}}_2 + \tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}(t - t_j),\ t \in [t_j, t_{j+1}[
\end{aligned} \tag{A.90}$$

which can then be solved one at a time for the transformed variables. Solving the equation for $\boldsymbol{z}_t^2$, with the initial condition $\boldsymbol{z}_{t=t_j}^2 = \boldsymbol{z}_j^2$, yields:

$$\boldsymbol{z}_t^2 = \boldsymbol{z}_j^2 + \tilde{\boldsymbol{f}}_2(t - t_j) + \frac{1}{2}\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}(t - t_j)^2\ ,\ t \in [t_j, t_{j+1}[\tag{A.91}$$

which can then be substituted into the equation for $\boldsymbol{z}_t^1$ to yield:

$$\begin{aligned}
\frac{d\boldsymbol{z}_t^1}{dt} = \tilde{\boldsymbol{f}}_1 &+ \tilde{\boldsymbol{A}}_1(\boldsymbol{z}_t^1 - \boldsymbol{z}_j^1) + \tilde{\boldsymbol{A}}_2\left(\tilde{\boldsymbol{f}}_2(t - t_j) + \frac{1}{2}\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}(t - t_j)^2\right) \\
&+ \tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}(t - t_j)\ ,\ t \in [t_j, t_{j+1}[
\end{aligned} \tag{A.92}$$

Introducing, for ease of notation, the constants:

$$\boldsymbol{E} = \frac{1}{2}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\ ,\ \boldsymbol{F} = \tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2 + \tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\ ,\ \boldsymbol{G} = \tilde{\boldsymbol{f}}_1 - \tilde{\boldsymbol{A}}_1\boldsymbol{z}_j^1 \tag{A.93}$$

and the standard form of a linear inhomogenous ordinary differential equation:

$$\frac{d\boldsymbol{z}_t^1}{dt} - \tilde{\boldsymbol{A}}_1\boldsymbol{z}_t^1 = \boldsymbol{E}(t - t_j)^2 + \boldsymbol{F}(t - t_j) + \boldsymbol{G}\ ,\ t \in [t_j, t_{j+1}[\tag{A.94}$$

gives the solution:

$$\boldsymbol{z}_t^1 = e^{\tilde{\boldsymbol{A}}_1 t}\left(\int e^{-\tilde{\boldsymbol{A}}_1 t}\left(\boldsymbol{E}(t-t_j)^2+\boldsymbol{F}(t-t_j)+\boldsymbol{G}\right)dt+\boldsymbol{c}\right),\ t\in[t_j,t_{j+1}[\tag{A.95}$$

which can be rearranged to:

$$\begin{aligned}\boldsymbol{z}_t^1 = &-\tilde{\boldsymbol{A}}_1^{-1}\left(\boldsymbol{I}(t-t_j)^2+2\tilde{\boldsymbol{A}}_1^{-1}(t-t_j)+2\tilde{\boldsymbol{A}}_1^{-2}\right)\boldsymbol{E}\\ &-\tilde{\boldsymbol{A}}_1^{-1}\left(\left(\boldsymbol{I}(t-t_j)+\tilde{\boldsymbol{A}}_1^{-1}\right)\boldsymbol{F}+\boldsymbol{G}\right)+e^{\tilde{\boldsymbol{A}}_1 t}\boldsymbol{c}\ ,\ t\in[t_j,t_{j+1}[\end{aligned} \tag{A.96}$$

Using the initial condition $\boldsymbol{z}^1_{t=t_j}=\boldsymbol{z}^1_j$ to determine the constant $\boldsymbol{c}$, i.e.:

$$\begin{aligned}\boldsymbol{z}_j^1 &= -\tilde{\boldsymbol{A}}_1^{-1}\left(2\tilde{\boldsymbol{A}}_1^{-2}\boldsymbol{E}+\tilde{\boldsymbol{A}}_1^{-1}\boldsymbol{F}+\boldsymbol{G}\right)+e^{\tilde{\boldsymbol{A}}_1 t_j}\boldsymbol{c}\\ \boldsymbol{c} &= e^{-\tilde{\boldsymbol{A}}_1 t_j}\left(\tilde{\boldsymbol{A}}_1^{-1}\left(2\tilde{\boldsymbol{A}}_1^{-2}\boldsymbol{E}+\tilde{\boldsymbol{A}}_1^{-1}\boldsymbol{F}+\boldsymbol{G}\right)+\boldsymbol{z}_j^1\right)\end{aligned} \tag{A.97}$$

the solution can be rearranged to:

$$\begin{aligned}\boldsymbol{z}_t^1 = &-\tilde{\boldsymbol{A}}_1^{-1}\left(\boldsymbol{I}(t-t_j)^2+2\tilde{\boldsymbol{A}}_1^{-1}(t-t_j)+2\tilde{\boldsymbol{A}}_1^{-2}\right)\boldsymbol{E}\\ &-\tilde{\boldsymbol{A}}_1^{-1}\left(\left(\boldsymbol{I}(t-t_j)+\tilde{\boldsymbol{A}}_1^{-1}\right)\boldsymbol{F}+\boldsymbol{G}\right)\\ &+e^{\tilde{\boldsymbol{A}}_1(t-t_j)}\left(\tilde{\boldsymbol{A}}_1^{-1}\left(2\tilde{\boldsymbol{A}}_1^{-2}\boldsymbol{E}+\tilde{\boldsymbol{A}}_1^{-1}\boldsymbol{F}+\boldsymbol{G}\right)+\boldsymbol{z}_j^1\right),\ t\in[t_j,t_{j+1}[\end{aligned} \tag{A.98}$$

which finally yields:

$$\begin{aligned}\boldsymbol{z}_{j+1}^1 &= -\tilde{\boldsymbol{A}}_1^{-1}\left(\left(\boldsymbol{I}\tau_s^2+2\tilde{\boldsymbol{A}}_1^{-1}\tau_s+2\tilde{\boldsymbol{A}}_1^{-2}\right)\boldsymbol{E}+\left(\boldsymbol{I}\tau_s+\tilde{\boldsymbol{A}}_1^{-1}\right)\boldsymbol{F}+\boldsymbol{G}\right)\\ &\quad+\tilde{\boldsymbol{\Phi}}_s^1\left(\tilde{\boldsymbol{A}}_1^{-1}\left(2\tilde{\boldsymbol{A}}_1^{-2}\boldsymbol{E}+\tilde{\boldsymbol{A}}_1^{-1}\boldsymbol{F}+\boldsymbol{G}\right)+\boldsymbol{z}_j^1\right)\\ &= -\tilde{\boldsymbol{A}}_1^{-1}\left(\left(\boldsymbol{I}\tau_s^2+2\tilde{\boldsymbol{A}}_1^{-1}\tau_s+2\tilde{\boldsymbol{A}}_1^{-2}\right)\frac{1}{2}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\right)\\ &\quad-\tilde{\boldsymbol{A}}_1^{-1}\left(\left(\boldsymbol{I}\tau_s+\tilde{\boldsymbol{A}}_1^{-1}\right)\left(\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2+\tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right)+\left(\tilde{\boldsymbol{f}}_1-\tilde{\boldsymbol{A}}_1\boldsymbol{z}_j^1\right)\right)\\ &\quad+\tilde{\boldsymbol{\Phi}}_s^1\left(\tilde{\boldsymbol{A}}_1^{-1}\left(2\tilde{\boldsymbol{A}}_1^{-2}\frac{1}{2}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}+\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2+\tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right)\right)\right)\\ &\quad+\tilde{\boldsymbol{\Phi}}_s^1\left(\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{f}}_1-\tilde{\boldsymbol{A}}_1\boldsymbol{z}_j^1\right)+\boldsymbol{z}_j^1\right)\\ &= \boldsymbol{z}_j^1-\tilde{\boldsymbol{A}}_1^{-1}\left(\frac{1}{2}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\tau_s^2+\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}+\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2+\tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right)\tau_s\right)\\ &\quad+\left(\tilde{\boldsymbol{\Phi}}_s^1-\boldsymbol{I}\right)\tilde{\boldsymbol{A}}_1^{-2}\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}+\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2+\tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}+\tilde{\boldsymbol{A}}_1\tilde{\boldsymbol{f}}_1\right)\\ &= \boldsymbol{z}_j^1-\frac{1}{2}\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\tau_s^2-\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}+\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2+\tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right)\tau_s\\ &\quad+\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{\Phi}}_s^1-\boldsymbol{I}\right)\left(\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}+\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2+\tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right)+\tilde{\boldsymbol{f}}_1\right)\end{aligned} \tag{A.99}$$

and:

$$\boldsymbol{z}_{j+1}^2 = \boldsymbol{z}_j^2 + \tilde{\boldsymbol{f}}_2\tau_s + \frac{1}{2}\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\tau_s^2 \tag{A.100}$$

where $\tilde{\boldsymbol{\Phi}}_s^1$ is the upper left part of the matrix:

$$\tilde{\boldsymbol{\Phi}}_s = \boldsymbol{U}^T\boldsymbol{\Phi}_s\boldsymbol{U} = \begin{bmatrix} \tilde{\boldsymbol{\Phi}}_s^1 & \tilde{\boldsymbol{\Phi}}_s^2 \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \tag{A.101}$$

and where the desired solution in terms of the original variables $\hat{\boldsymbol{x}}_{j+1|j}$ can be found by applying the reverse transformation $\hat{\boldsymbol{x}}_t = \boldsymbol{U}\boldsymbol{z}_t$.

Depending on the specific singularity of $\boldsymbol{A}$ (see Section A.1.3.3 for details on how this is determined in **CTSM**) and the particular nature of the inputs, several different cases are possible as shown in the following.

General case: Singular $\boldsymbol{A}$, first order hold on inputs

In the general case, the extended Kalman filter solution is given as follows:

$$\begin{aligned} \boldsymbol{z}_{j+1|j}^1 = \boldsymbol{z}_{j|j}^1 &- \frac{1}{2}\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\tau_s^2 \\ &- \tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha} + \tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2 + \tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right)\tau_s \\ &+ \tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I}\right)\left(\tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha} + \tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2 + \tilde{\boldsymbol{B}}_1\boldsymbol{\alpha}\right) + \tilde{\boldsymbol{f}}_1\right) \end{aligned} \tag{A.102}$$

and:

$$\boldsymbol{z}_{j+1|j}^2 = \boldsymbol{z}_{j|j}^2 + \tilde{\boldsymbol{f}}_2\tau_s + \frac{1}{2}\tilde{\boldsymbol{B}}_2\boldsymbol{\alpha}\tau_s^2 \tag{A.103}$$

where the desired solution in terms of the original variables $\hat{\boldsymbol{x}}_{j+1|j}$ can be found by applying the reverse transformation $\hat{\boldsymbol{x}}_t = \boldsymbol{U}\boldsymbol{z}_t$.

Special case no. 1: Singular $\boldsymbol{A}$, zero order hold on inputs

The solution to this special case can be obtained by setting $\boldsymbol{\alpha} = 0$, which yields:

$$\boldsymbol{z}_{j+1|j}^1 = \boldsymbol{z}_{j|j}^1 - \tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2\tau_s + \tilde{\boldsymbol{A}}_1^{-1}\left(\tilde{\boldsymbol{\Phi}}_s^1 - \boldsymbol{I}\right)\left(\tilde{\boldsymbol{A}}_1^{-1}\tilde{\boldsymbol{A}}_2\tilde{\boldsymbol{f}}_2 + \tilde{\boldsymbol{f}}_1\right) \tag{A.104}$$

and:

$$\boldsymbol{z}_{j+1|j}^2 = \boldsymbol{z}_{j|j}^2 + \tilde{\boldsymbol{f}}_2\tau_s \tag{A.105}$$

where the desired solution in terms of the original variables $\hat{\boldsymbol{x}}_{j+1|j}$ can be found by applying the reverse transformation $\hat{\boldsymbol{x}}_t = \boldsymbol{U}\boldsymbol{z}_t$.

Special case no. 2: Nonsingular $\boldsymbol{A}$, first order hold on inputs

The solution to this special case can be obtained by removing the SVD dependent parts, i.e. by replacing $\boldsymbol{z}_t^1$, $\tilde{\boldsymbol{A}}_1$, $\tilde{\boldsymbol{B}}_1$ and $\tilde{\boldsymbol{f}}_1$ with $\boldsymbol{x}_t$, $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{f}$ respectively, and by setting $\boldsymbol{z}_t^2$, $\tilde{\boldsymbol{A}}_2$, $\tilde{\boldsymbol{B}}_2$ and $\tilde{\boldsymbol{f}}_2$ to zero, which yields:

$$\hat{\boldsymbol{x}}_{j+1|j} = \hat{\boldsymbol{x}}_{j|j} - \boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{\alpha}\tau_s + \boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right)\left(\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{\alpha} + \boldsymbol{f}\right) \tag{A.106}$$

Special case no. 3: Nonsingular $\boldsymbol{A}$, zero order hold on inputs

The solution to this special case can be obtained by removing the SVD dependent parts, i.e. by replacing $\boldsymbol{z}_t^1$, $\tilde{\boldsymbol{A}}_1$, $\tilde{\boldsymbol{B}}_1$ and $\tilde{\boldsymbol{f}}_1$ with $\boldsymbol{x}_t$, $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{f}$ respectively, and by setting $\boldsymbol{z}_t^2$, $\tilde{\boldsymbol{A}}_2$, $\tilde{\boldsymbol{B}}_2$ and $\tilde{\boldsymbol{f}}_2$ to zero and $\boldsymbol{\alpha} = 0$, which yields:

$$\hat{\boldsymbol{x}}_{j+1|j} = \hat{\boldsymbol{x}}_{j|j} + \boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right)\boldsymbol{f} \tag{A.107}$$

Special case no. 4: Identically zero $\boldsymbol{A}$, first order hold on inputs

The solution to this special case can be obtained by setting $\boldsymbol{A}$ to zero and solving the original linearized state propagation equation, which yields:

$$\hat{\boldsymbol{x}}_{j+1|j} = \hat{\boldsymbol{x}}_{j|j} + \boldsymbol{f}\tau_s + \frac{1}{2}\boldsymbol{B}\boldsymbol{\alpha}\tau_s^2 \tag{A.108}$$

Special case no. 5: Identically zero $\boldsymbol{A}$, zero order hold on inputs

The solution to this special case can be obtained by setting $\boldsymbol{A}$ to zero and $\boldsymbol{\alpha} = 0$ and solving the original linearized state propagation equation, which yields:

$$\hat{\boldsymbol{x}}_{j+1|j} = \hat{\boldsymbol{x}}_{j|j} + \boldsymbol{f}\tau_s \tag{A.109}$$

Numerical ODE solution as an alternative

The subsampling-based solution framework described above provides a better approximation to the true state propagation solution than direct numerical solution of (A.78) and (A.79), because it more accurately reflects the true time-varying nature of the matrices $\boldsymbol{A}$ and $\boldsymbol{\sigma}$ in (A.79) by allowing these to be re-evaluated at each subsampling instant. To provide an even better approximation and to handle stiff systems, which is not always possible with the subsampling-based solution framework, an option has been included in **CTSM** for applying numerical ODE solution to solve (A.78) and (A.79) simultaneously[6], which ensures intelligent re-evaluation of $\boldsymbol{A}$ and $\boldsymbol{\sigma}$ in (A.79).

[6]The specific implementation is based on the algorithms of Hindmarsh (1983), and to be able to use this method to solve (A.78) and (A.79) simultaneously, the n-vector differential equation in (A.78) has been augmented with an $n \times (n+1)/2$-vector differential equation corresponding to the symmetric $n \times n$-matrix differential equation in (A.79).

Iterated extended Kalman filtering

The sensitivity of the extended Kalman filter to nonlinear effects not only means that the approximation to the true state propagation solution provided by the solution to the state prediction equations (A.78) and (A.79) may be too crude. The presence of such effects in the output prediction equations (A.72) and (A.73) may also influence the performance of the filter. An option has therefore been included in **CTSM** for applying the *iterated extended Kalman filter* (Jazwinski, 1970), which is an iterative version of the extended Kalman filter that consists of the modified output prediction equations:

$$\hat{\boldsymbol{y}}^i_{k|k-1} = \boldsymbol{h}(\eta_i, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \tag{A.110}$$

$$\boldsymbol{R}^i_{k|k-1} = \boldsymbol{C}_i \boldsymbol{P}_{k|k-1} \boldsymbol{C}_i^T + \boldsymbol{S} \tag{A.111}$$

the modified innovation equation:

$$\boldsymbol{\epsilon}^i_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}^i_{k|k-1} \tag{A.112}$$

the modified Kalman gain equation:

$$\boldsymbol{K}^i_k = \boldsymbol{P}_{k|k-1} \boldsymbol{C}_i^T (\boldsymbol{R}^i_{k|k-1})^{-1} \tag{A.113}$$

and the modified updating equations:

$$\eta_{i+1} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{\epsilon}^i_k - \boldsymbol{C}_i(\hat{\boldsymbol{x}}_{k|k-1} - \eta_i)) \tag{A.114}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}^i_k \boldsymbol{R}^i_{k|k-1} (\boldsymbol{K}^i_k)^T \tag{A.115}$$

where:

$$\boldsymbol{C}_i = \left. \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_t} \right|_{\boldsymbol{x}=\eta_i, \boldsymbol{u}=\boldsymbol{u}_k, t=t_k, \boldsymbol{\theta}} \tag{A.116}$$

and $\eta_1 = \hat{\boldsymbol{x}}_{k|k-1}$. The above equations are iterated for $i = 1, \ldots, M$, where M is the maximum number of iterations, or until there is no significant difference between consecutive iterates, whereupon $\hat{\boldsymbol{x}}_{k|k} = \eta_M$ is assigned. This way, the influence of nonlinear effects in (A.72) and (A.73) can be reduced.

A.1.3.3 Determination of singularity

Computing the singular value decomposition (SVD) of a matrix is a computationally expensive task, which should be avoided if possible. Within **CTSM** the determination of whether or not the $\boldsymbol{A}$ matrix is singular and thus whether or not the SVD should be applied, therefore is not based on the SVD itself, but on an estimate of the reciprocal condition number, i.e.:

$$\hat{\kappa}^{-1} = \frac{1}{|\boldsymbol{A}||\boldsymbol{A}^{-1}|} \tag{A.117}$$

where $|\boldsymbol{A}|$ is the 1-norm of the $\boldsymbol{A}$ matrix and $|\boldsymbol{A}^{-1}|$ is an estimate of the 1-norm of $\boldsymbol{A}^{-1}$. This quantity can be computed much faster than the SVD, and only if its value is below a certain threshold (e.g. 1e-12), the SVD is applied.

A.1.3.4 Initial states and covariances

In order for the (extended) Kalman filter to work, the initial states $\boldsymbol{x}_0$ and their covariance matrix $\boldsymbol{P}_0$ must be specified. Within **CTSM** the initial states may either be pre-specified or estimated by the program along with the parameters, whereas the initial covariance matrix is calculated as $\boldsymbol{P}_0 = P_s \boldsymbol{\sigma}\boldsymbol{\sigma}^T$, where $\boldsymbol{\sigma}$ corresponds to the first sample and P_s is a pre-specified scaling factor.

A.1.3.5 Factorization of covariance matrices

The (extended) Kalman filter may be numerically unstable in certain situations. The problem arises when some of the covariance matrices, which are known from theory to be symmetric and positive definite, become non-positive definite because of rounding errors. Consequently, careful handling of the covariance equations is needed to stabilize the (extended) Kalman filter. Within **CTSM**, all covariance matrices are therefore replaced with their square root free Cholesky decompositions (Fletcher and Powell, 1974), i.e.:

$$\boldsymbol{P} = \boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^T \tag{A.118}$$

where $\boldsymbol{P}$ is the covariance matrix, $\boldsymbol{L}$ is a unit lower triangular matrix and $\boldsymbol{D}$ is a diagonal matrix with $d_{ii} > 0$, $\forall i$. Using factorized covariance matrices, all of the covariance equations of the (extended) Kalman filter can be handled by means of the following equation for updating a factorized matrix:

$$\tilde{\boldsymbol{P}} = \boldsymbol{P} + \boldsymbol{G}\boldsymbol{D}_g\boldsymbol{G}^T \tag{A.119}$$

where $\tilde{\boldsymbol{P}}$ is known from theory to be both symmetric and positive definite and $\boldsymbol{P}$ is given by (A.118), and where $\boldsymbol{D}_g$ is a diagonal matrix and $\boldsymbol{G}$ is a full matrix. Solving this equation amounts to finding a unit lower triangular matrix $\tilde{\boldsymbol{L}}$ and a diagonal matrix $\tilde{\boldsymbol{D}}$ with $\tilde{d}_{ii} > 0$, $\forall i$, such that:

$$\tilde{\boldsymbol{P}} = \tilde{\boldsymbol{L}}\tilde{\boldsymbol{D}}\tilde{\boldsymbol{L}}^T \tag{A.120}$$

and for this purpose a number of different methods are available, e.g. the method described by Fletcher and Powell (1974), which is based on the modified Givens transformation, and the method described by Thornton and Bierman (1980), which is based on the modified weighted Gram-Schmidt orthogonalization. Within **CTSM** the specific implementation of the (extended) Kalman filter is based on the latter, and this implementation has been proven to have a high grade of accuracy as well as stability (Bierman, 1977).

Using factorized covariance matrices also facilitates easy computation of those parts of the objective function (A.26) that depend on determinants of covariance matrices. This is due to the following identities:

$$\det(\boldsymbol{P}) = \det(\boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^T) = \det(\boldsymbol{D}) = \prod_i d_{ii} \tag{A.121}$$

A.1.4 Data issues

Raw data sequences are often difficult to use for identification and parameter estimation purposes, e.g. if irregular sampling has been applied, if there are occasional outliers or if some of the observations are missing. **CTSM** also provides features to deal with these issues, and this makes the program flexible with respect to the types of data that can be used for the estimation.

A.1.4.1 Irregular sampling.

The fact that the system equation of a continuous-discrete stochastic state space model is formulated in continuous time makes it easy to deal with irregular sampling, because the corresponding state prediction equations of the (extended) Kalman filter can be solved over time intervals of varying length.

A.1.4.2 Occasional outliers

The objective function (A.26) of the general formulation (A.27) is quadratic in the innovations $\boldsymbol{\epsilon}_k^i$, and this means that the corresponding parameter estimates are heavily influenced by occasional outliers in the data sets used for the estimation. To deal with this problem, a robust estimation method is applied, where the objective function is modified by replacing the quadratic term:

$$\nu_k^i = (\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i \tag{A.122}$$

with a threshold function $\varphi(\nu_k^i)$, which returns the argument for small values of ν_k^i, but is a linear function of $\boldsymbol{\epsilon}_k^i$ for large values of ν_k^i, i.e.:

$$\varphi(\nu_k^i) = \begin{cases} \nu_k^i & , \ \nu_k^i < c^2 \\ c(2\sqrt{\nu_k^i} - c) & , \ \nu_k^i \geq c^2 \end{cases} \tag{A.123}$$

where $c > 0$ is a constant. The derivative of this function with respect to $\boldsymbol{\epsilon}_k^i$ is known as *Huber's ψ-function* (Huber, 1981) and belongs to a class of functions called influence functions, because they measure the influence of $\boldsymbol{\epsilon}_k^i$ on the objective function. Several such functions are available, but Huber's ψ-function has been found to be most appropriate in terms of providing robustness against outliers without rendering optimisation of the objective function infeasible.

A.1.4.3 Missing observations.

The algorithms of the parameter estimation methods described above also make it easy to handle missing observations, i.e. to account for missing values in the output vector $\boldsymbol{y}_k^i$, for some i and some k, when calculating the terms:

$$\frac{1}{2} \sum_{i=1}^{S} \sum_{k=1}^{N_i} \left(\ln(\det(\boldsymbol{R}_{k|k-1}^i)) + (\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i \right) \tag{A.124}$$

and:

$$\frac{1}{2}\left(\left(\sum_{i=1}^{S}\sum_{k=1}^{N_i} l\right) + p\right)\ln(2\pi) \tag{A.125}$$

in (A.26). To illustrate this, the case of extended Kalman filtering for NL models is considered, but similar arguments apply in the case of Kalman filtering for LTI and LTV models. The usual way to account for missing or non-informative values in the extended Kalman filter is to formally set the corresponding elements of the measurement error covariance matrix $\boldsymbol{S}$ in (A.73) to infinity, which in turn gives zeroes in the corresponding elements of the inverted output covariance matrix $(\boldsymbol{R}_{k|k-1})^{-1}$ and the Kalman gain matrix $\boldsymbol{K}_k$, meaning that no updating will take place in (A.76) and (A.77) corresponding to the missing values. This approach cannot be used when calculating (A.124) and (A.125), however, because a solution is needed which modifies both $\boldsymbol{\epsilon}_k^i$, $\boldsymbol{R}_{k|k-1}^i$ and l to reflect that the effective dimension of $\boldsymbol{y}_k^i$ is reduced. This is accomplished by replacing (A.2) with the alternative measurement equation:

$$\overline{\boldsymbol{y}}_k = \boldsymbol{E}\left(\boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k\right) \tag{A.126}$$

where $\boldsymbol{E}$ is an appropriate permutation matrix, which can be constructed from a unit matrix by eliminating the rows that correspond to the missing values in $\boldsymbol{y}_k$. If, for example, $\boldsymbol{y}_k$ has three elements, and the one in the middle is missing, the appropriate permutation matrix is given as follows:

$$\boldsymbol{E} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.127}$$

Equivalently, the equations of the extended Kalman filter are replaced with the following alternative output prediction equations:

$$\hat{\overline{\boldsymbol{y}}}_{k|k-1} = \boldsymbol{E}\boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \tag{A.128}$$

$$\overline{\boldsymbol{R}}_{k|k-1} = \boldsymbol{E}\boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{E}^T + \boldsymbol{E}\boldsymbol{S}\boldsymbol{E}^T \tag{A.129}$$

the alternative innovation equation:

$$\overline{\boldsymbol{\epsilon}}_k = \overline{\boldsymbol{y}}_k - \hat{\overline{\boldsymbol{y}}}_{k|k-1} \tag{A.130}$$

the alternative Kalman gain equation:

$$\overline{\boldsymbol{K}}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{E}^T\overline{\boldsymbol{R}}_{k|k-1}^{-1} \tag{A.131}$$

and the alternative updating equations:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \overline{\boldsymbol{K}}_k\overline{\boldsymbol{\epsilon}}_k \tag{A.132}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \overline{\boldsymbol{K}}_k\overline{\boldsymbol{R}}_{k|k-1}\overline{\boldsymbol{K}}_k^T \tag{A.133}$$

The state prediction equations remain the same, and the above replacements in turn provide the necessary modifications of (A.124) to:

$$\frac{1}{2}\sum_{i=1}^{S}\sum_{k=1}^{N_i}\left(\ln(\det(\overline{\boldsymbol{R}}^i_{k|k-1})) + (\overline{\boldsymbol{\epsilon}}^i_k)^T(\overline{\boldsymbol{R}}^i_{k|k-1})^{-1}\overline{\boldsymbol{\epsilon}}^i_k\right) \tag{A.134}$$

whereas modifying (A.125) amounts to a simple reduction of l for the particular values of i and k with the number of missing values in $\boldsymbol{y}^i_k$.

A.1.5 Optimisation issues

CTSM uses a *quasi-Newton* method based on the BFGS updating formula and a soft line search algorithm to solve the nonlinear optimisation problem (A.27). This method is similar to the one described by Dennis and Schnabel (1983), except for the fact that the gradient of the objective function is approximated by a set of finite difference derivatives. In analogy with ordinary Newton-Raphson methods for optimisation, quasi-Newton methods seek a minimum of a nonlinear objective function $\mathcal{F}(\boldsymbol{\theta})$: $\mathbb{R}^p \to \mathbb{R}$, i.e.:

$$\min_{\boldsymbol{\theta}} \mathcal{F}(\boldsymbol{\theta}) \tag{A.135}$$

where a minimum of $\mathcal{F}(\boldsymbol{\theta})$ is found when the gradient $\boldsymbol{g}(\boldsymbol{\theta}) = \frac{\partial \mathcal{F}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$ satisfies:

$$\boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{0} \tag{A.136}$$

Both types of methods are based on the Taylor expansion of $\boldsymbol{g}(\boldsymbol{\theta})$ to first order:

$$\boldsymbol{g}(\boldsymbol{\theta}^i + \boldsymbol{\delta}) = \boldsymbol{g}(\boldsymbol{\theta}^i) + \frac{\partial \boldsymbol{g}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}^i}\,\boldsymbol{\delta} + o(\boldsymbol{\delta}) \tag{A.137}$$

which by setting $\boldsymbol{g}(\boldsymbol{\theta}^i + \boldsymbol{\delta}) = \boldsymbol{0}$ and neglecting $o(\boldsymbol{\delta})$ can be rewritten as follows:

$$\boldsymbol{\delta}^i = -\boldsymbol{H}_i^{-1}\boldsymbol{g}(\boldsymbol{\theta}^i) \tag{A.138}$$

$$\boldsymbol{\theta}^{i+1} = \boldsymbol{\theta}^i + \boldsymbol{\delta}^i \tag{A.139}$$

i.e. as an iterative algorithm, and this algorithm can be shown to converge to a (possibly local) minimum. The Hessian $\boldsymbol{H}_i$ is defined as follows:

$$\boldsymbol{H}_i = \frac{\partial \boldsymbol{g}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}^i} \tag{A.140}$$

but unfortunately neither the Hessian nor the gradient can be computed explicitly for the optimisation problem (A.27). As mentioned above, the gradient is therefore approximated by a set of finite difference derivatives, and a secant approximation based on the BFGS updating formula is applied for the Hessian. It is the use of a secant approximation to the Hessian that distinguishes quasi-Newton methods from ordinary Newton-Raphson methods.

A.1.5.1 Finite difference derivative approximations

Since the gradient $\boldsymbol{g}(\boldsymbol{\theta}^i)$ cannot be computed explicitly, it is approximated by a set of finite difference derivatives. Initially, i.e. as long as $||\boldsymbol{g}(\boldsymbol{\theta})||$ does not become too small during the iterations of the optimisation algorithm, *forward difference* approximations are used, i.e.:

$$g_j(\boldsymbol{\theta}^i) \approx \frac{\mathcal{F}(\boldsymbol{\theta}^i + \delta_j \boldsymbol{e}_j) - \mathcal{F}(\boldsymbol{\theta}^i)}{\delta_j} \ , \ j = 1, \ldots, p \tag{A.141}$$

where $g_j(\boldsymbol{\theta}^i)$ is the j'th component of $\boldsymbol{g}(\boldsymbol{\theta}^i)$ and $\boldsymbol{e}_j$ is the j'th basis vector. The error of this type of approximation is $o(\delta_j)$. Subsequently, i.e. when $||\boldsymbol{g}(\boldsymbol{\theta})||$ becomes small near a minimum of the objective function, *central difference* approximations are used instead, i.e.:

$$g_j(\boldsymbol{\theta}^i) \approx \frac{\mathcal{F}(\boldsymbol{\theta}^i + \delta_j \boldsymbol{e}_j) - \mathcal{F}(\boldsymbol{\theta}^i - \delta_j \boldsymbol{e}_j)}{2\delta_j} \ , \ j = 1, \ldots, p \tag{A.142}$$

because the error of this type of approximation is only $o(\delta_j^2)$. Unfortunately, central difference approximations require twice as much computation (twice the number of objective function evalutions) as forward difference approximations, so to save computation time forward difference approximations are used initially. The switch from forward differences to central differences is effectuated for $i > 2p$ if the line search algorithm fails to find a better value of $\boldsymbol{\theta}$.

The optimal choice of step length for forward difference approximations is:

$$\delta_j = \eta^{\frac{1}{2}} \theta_j \tag{A.143}$$

whereas for central difference approximations it is:

$$\delta_j = \eta^{\frac{1}{3}} \theta_j \tag{A.144}$$

where η is the relative error of calculating $\mathcal{F}(\boldsymbol{\theta})$ (Dennis and Schnabel, 1983).

A.1.5.2 The BFGS updating formula

Since the Hessian $\boldsymbol{H}_i$ cannot be computed explicitly, a secant approximation is applied. The most effective secant approximation $\boldsymbol{B}_i$ is obtained with the so-called BFGS updating formula (Dennis and Schnabel, 1983), i.e.:

$$\boldsymbol{B}_{i+1} = \boldsymbol{B}_i + \frac{\boldsymbol{y}_i \boldsymbol{y}_i^T}{\boldsymbol{y}_i^T \boldsymbol{s}_i} - \frac{\boldsymbol{B}_i \boldsymbol{s}_i \boldsymbol{s}_i^T \boldsymbol{B}_i}{\boldsymbol{s}_i^T \boldsymbol{B}_i \boldsymbol{s}_i} \tag{A.145}$$

where $\boldsymbol{y}_i = \boldsymbol{g}(\boldsymbol{\theta}_{i+1}) - \boldsymbol{g}(\boldsymbol{\theta}_i)$ and $\boldsymbol{s}_i = \boldsymbol{\theta}_{i+1} - \boldsymbol{\theta}_i$. Necessary and sufficient conditions for $\boldsymbol{B}_{i+1}$ to be positive definite is that $\boldsymbol{B}_i$ is positive definite and that:

$$\boldsymbol{y}_i^T \boldsymbol{s}_i > 0 \tag{A.146}$$

This last demand is automatically met by the line search algorithm. Furthermore, since the Hessian is symmetric and positive definite, it can also be written in terms of its square root free Cholesky factors, i.e.:

$$\boldsymbol{B}_i = \boldsymbol{L}_i \boldsymbol{D}_i \boldsymbol{L}_i^T \tag{A.147}$$

where $\boldsymbol{L}_i$ is a unit lower triangular matrix and $\boldsymbol{D}_i$ is a diagonal matrix with $d_{jj}^i > 0$, $\forall j$, so, instead of solving (A.145) directly, $\boldsymbol{B}_{i+1}$ can be found by updating the Cholesky factorization of $\boldsymbol{B}_i$ as shown in Section A.1.3.5.

A.1.5.3 The soft line search algorithm

With $\boldsymbol{\delta}^i$ being the secant direction from (A.138) (using $\boldsymbol{H}_i = \boldsymbol{B}_i$ obtained from (A.145)), the idea of the soft line search algorithm is to replace (A.139) with:

$$\boldsymbol{\theta}^{i+1} = \boldsymbol{\theta}^i + \lambda_i \boldsymbol{\delta}^i \tag{A.148}$$

and choose a value of $\lambda_i > 0$ that ensures that the next iterate decreases $\mathcal{F}(\boldsymbol{\theta})$ and that (A.146) is satisfied. Often $\lambda_i = 1$ will satisfy these demands and (A.148) reduces to (A.139). The soft line search algorithm is globally convergent if each step satisfies two simple conditions. The first condition is that the decrease in $\mathcal{F}(\boldsymbol{\theta})$ is sufficient compared to the length of the step $\boldsymbol{s}_i = \lambda_i \boldsymbol{\delta}^i$, i.e.:

$$\mathcal{F}(\boldsymbol{\theta}^{i+1}) < \mathcal{F}(\boldsymbol{\theta}^i) + \alpha \boldsymbol{g}(\boldsymbol{\theta}^i)^T \boldsymbol{s}_i \tag{A.149}$$

where $\alpha \in]0, 1[$. The second condition is that the step is not too short, i.e.:

$$\boldsymbol{g}(\boldsymbol{\theta}^{i+1})^T \boldsymbol{s}_i \geq \beta \boldsymbol{g}(\boldsymbol{\theta}^i)^T \boldsymbol{s}_i \tag{A.150}$$

where $\beta \in]\alpha, 1[$. This last expression and $\boldsymbol{g}(\boldsymbol{\theta}^i)^T \boldsymbol{s}_i < 0$ imply that:

$$\boldsymbol{y}_i^T \boldsymbol{s}_i = \left(\boldsymbol{g}(\boldsymbol{\theta}^{i+1}) - \boldsymbol{g}(\boldsymbol{\theta}^i)\right)^T \boldsymbol{s}_i \geq (\beta - 1) \boldsymbol{g}(\boldsymbol{\theta}^i)^T \boldsymbol{s}_i > 0 \tag{A.151}$$

which guarantees that (A.146) is satisfied. The method for finding a value of λ_i that satisfies both (A.149) and (A.150) starts out by trying $\lambda_i = \lambda_p = 1$. If this trial value is not admissible because it fails to satisfy (A.149), a decreased value is found by cubic interpolation using $\mathcal{F}(\boldsymbol{\theta}^i)$, $\boldsymbol{g}(\boldsymbol{\theta}^i)$, $\mathcal{F}(\boldsymbol{\theta}^i + \lambda_p \boldsymbol{\delta}^i)$ and $\boldsymbol{g}(\boldsymbol{\theta}^i + \lambda_p \boldsymbol{\delta}^i)$. If the trial value satisfies (A.149) but not (A.150), an increased value is found by extrapolation. After one or more repetitions, an admissible λ_i is found, because it can be proved that there exists an interval $\lambda_i \in [\lambda_1, \lambda_2]$ where (A.149) and (A.150) are both satisfied (Dennis and Schnabel, 1983).

A.1.5.4 Constraints on parameters

In order to ensure stability in the calculation of the objective function in (A.26), simple constraints on the parameters are introduced, i.e.:

$$\theta_j^{\min} < \theta_j < \theta_j^{\max} \ , \ j = 1, \ldots, p \tag{A.152}$$

These constraints are satisfied by solving the optimisation problem with respect to a transformation of the original parameters, i.e.:

$$\tilde{\theta}_j = \ln\left(\frac{\theta_j - \theta_j^{\min}}{\theta_j^{\max} - \theta_j}\right) \ , j = 1, \ldots, p \tag{A.153}$$

A problem arises with this type of transformation when θ_j is very close to one of the limits, because the finite difference derivative with respect to θ_j may be close to zero, but this problem is solved by adding an appropriate penalty function to (A.26) to give the following modified objective function:

$$\mathcal{F}(\boldsymbol{\theta}) = -\ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right) + P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max}) \tag{A.154}$$

which is then used instead. The penalty function is given as follows:

$$P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max}) = \lambda\left(\sum_{j=1}^{p} \frac{|\theta_j^{\min}|}{\theta_j - \theta_j^{\min}} + \sum_{j=1}^{p} \frac{|\theta_j^{\max}|}{\theta_j^{\max} - \theta_j}\right) \tag{A.155}$$

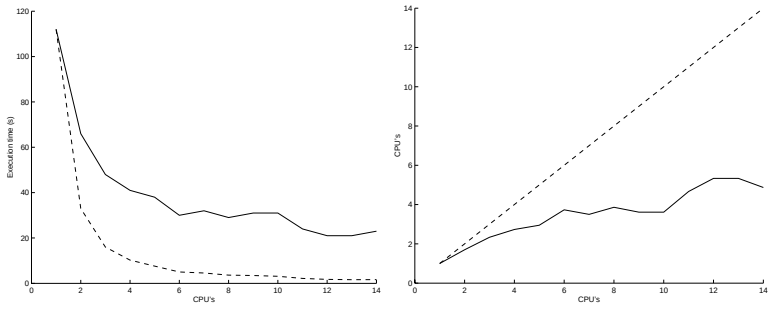
for $|\theta_j^{\min}| > 0$ and $|\theta_j^{\max}| > 0$, $j = 1, \ldots, p$. For proper choices of the Lagrange multiplier λ and the limiting values $\theta_j^{\min}$ and $\theta_j^{\max}$ the penalty function has no influence on the estimation when θ_j is well within the limits but will force the finite difference derivative to increase when θ_j is close to one of the limits.

Along with the parameter estimates **CTSM** computes normalized (by multiplication with the estimates) derivatives of $\mathcal{F}(\boldsymbol{\theta})$ and $P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max})$ with respect to the parameters to provide information about the solution. The derivatives of $\mathcal{F}(\boldsymbol{\theta})$ should of course be close to zero, and the absolute values of the derivatives of $P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max})$ should not be large compared to the corresponding absolute values of the derivatives of $\mathcal{F}(\boldsymbol{\theta})$, because this indicates that the corresponding parameters are close to one of their limits.

A.1.6 Performance issues

Solving optimisation problems of the general type in (A.27) is a computationally intensive task. The binary code within **CTSM** has therefore been optimized for maximum performance on all supported platforms, i.e. Linux, Solaris and Windows. On Solaris systems **CTSM** also supports shared memory parallel computing using the OpenMP Application Program Interface (API).

More specifically, the finite difference derivative approximations used to approximate the gradient of the objective function can be computed in parallel, and Figure A.1 shows the performance benefits of this approach in terms of reduced execution time and demonstrates the resulting scalability of the program for the bioreactor example used in Chapter 2. In this example there are 11 unknown parameters, and in theory using 11 CPU's should therefore be most optimal. Nevertheless, using 12 CPU's seems to be slightly better, but



(a) Performance. (b) Scalability.

Figure A.1. Performance (execution time vs. no. of CPU's) and scalability (no. of CPU's vs. no. of CPU's) of **CTSM** when using shared memory parallel computing. Solid lines: **CTSM** values; dashed lines: Theoretical values (linear scalability).

this may be due to the inherent uncertainty of the determination of execution time. The apparently non-existing effect of adding CPU's in the interval 6-10 is due to an uneven distribution of the workload, since in this case at least one CPU performs two finite difference computations, while the others wait.

A.2 Other features

Secondary features of **CTSM** include computation of various statistics and facilitation of residual analysis through validation data generation.

A.2.1 Various statistics

Within **CTSM** an estimate of the uncertainty of the parameter estimates is obtained by using the fact that by the central limit theorem the estimator in (A.27) is asymptotically Gaussian with mean $\boldsymbol{\theta}$ and covariance:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{H}^{-1} \tag{A.156}$$

where the matrix $\boldsymbol{H}$ is given by:

$$\{h_{ij}\} = -E\left\{\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln\left(p(\boldsymbol{\theta}|\mathbf{Y},\mathbf{y_0})\right)\right\}\ , i,j=1,\ldots,p \tag{A.157}$$

and where an approximation to $\boldsymbol{H}$ can be obtained from:

$$\{h_{ij}\} \approx -\left(\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln\left(p(\boldsymbol{\theta}|\mathbf{Y},\mathbf{y_0})\right)\right)\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}\ , i,j=1,\ldots,p \tag{A.158}$$

which is the Hessian evaluated at the minimum of the objective function, i.e. $\boldsymbol{H}_i|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}$. As an overall measure of the uncertainty of the parameter estimates, the negative logarithm of the determinant of the Hessian is computed, i.e.:

$$-\ln\left(\det\left(\boldsymbol{H}_i|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}\right)\right) \tag{A.159}$$

The lower the value of this statistic, the lower the overall uncertainty of the parameter estimates. A measure of the uncertainty of the individual parameter estimates is obtained by decomposing the covariance matrix as follows:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}}\boldsymbol{R}\boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}} \tag{A.160}$$

into $\boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}}$, which is a diagonal matrix of the standard deviations of the parameter estimates, and $\boldsymbol{R}$, which is the corresponding correlation matrix.

The asymptotic Gaussianity of the estimator in (A.27) also allows marginal t-tests to be performed to test the hypothesis:

$$H_0\text{:}\quad \theta_j = 0 \tag{A.161}$$

against the corresponding alternative:

$$H_1\text{:}\quad \theta_j \neq 0 \tag{A.162}$$

i.e. to test whether a given parameter θ_j is marginally insignificant or not. The test quantity is the value of the parameter estimate divided by the standard deviation of the estimate, and under H_0 this quantity is asymptotically t-distributed with a number of degrees of freedom DF that equals the total number of observations minus the number of estimated parameters, i.e.:

$$z^t(\hat{\theta}_j) = \frac{\hat{\theta}_j}{\sigma_{\hat{\theta}_j}} \quad \in \quad t(\mathrm{DF}) = t\left(\left(\sum_{i=1}^{S}\sum_{k=1}^{N_i} l\right) - p\right) \tag{A.163}$$

where, if there are missing observations in $\boldsymbol{y}_k^i$ for some i and some k, the particular value of l is reduced with the number of missing values in $\boldsymbol{y}_k^i$. The critical region for a test on significance level α is given as follows:

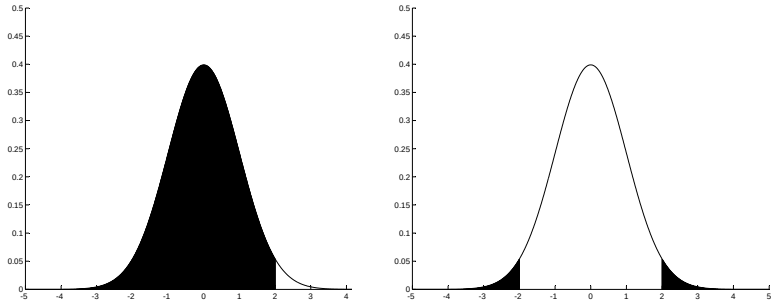
$$z^t(\hat{\theta}_j) < t(\mathrm{DF})_{\frac{\alpha}{2}} \ \vee \ z^t(\hat{\theta}_j) > t(\mathrm{DF})_{1-\frac{\alpha}{2}} \tag{A.164}$$

and to facilitate these tests, **CTSM** computes $z^t(\hat{\theta}_j)$ as well as the probabilities:

$$P\left(t<-|z^t(\hat{\theta}_j)| \wedge t>|z^t(\hat{\theta}_j)|\right) \tag{A.165}$$

for $j = 1, \ldots, p$. Figure A.2 shows how these probabilities should be interpreted and illustrates their computation via the following relation:

$$P\left(t<-|z^t(\hat{\theta}_j)| \wedge t>|z^t(\hat{\theta}_j)|\right) = 2\left(1 - P(t < |z^t(\hat{\theta}_j)|)\right) \tag{A.166}$$



(a) $P(t < |z^t(\hat{\theta}_j)|)$. (b) $P(t<-|z^t(\hat{\theta}_j)| \wedge t>|z^t(\hat{\theta}_j)|)$.

Figure A.2. Illustration of computation of $P(t<-|z^t(\hat{\theta}_j)| \wedge t>|z^t(\hat{\theta}_j)|)$ via (A.166).

with $P(t < |z^t(\hat{\theta}_j)|)$ obtained by approximating the cumulative probability density of the t-distribution $t(\mathrm{DF})$ with the cumulative probability density of the standard Gaussian distribution $N(0,1)$ using the test quantity transformation:

$$z^N(\hat{\theta}_j) = z^t(\hat{\theta}_j)\frac{1-\frac{1}{4\mathrm{DF}}}{\sqrt{1+\frac{(z^t(\hat{\theta}_j))^2}{2\mathrm{DF}}}} \quad \in \quad N(0,1) \tag{A.167}$$

The cumulative probability density of the standard Gaussian distribution is computed by approximation using a series expansion of the error function.

A.2.2 Validation data generation

To facilitate e.g. residual analysis, **CTSM** can also be used to generate validation data, i.e. state and output predictions corresponding to a given input data set, using either one-step-ahead prediction or pure simulation.

A.2.2.1 One-step-ahead prediction data generation

The one-step-ahead state and output predictions that can be generated are $\hat{\boldsymbol{x}}_{k|k-1}$, $\hat{\boldsymbol{x}}_{k|k}$ and $\hat{\boldsymbol{y}}_{k|k-1}$ corresponding to each time instant t_k in the input data set. The predictions are generated by the (extended) Kalman filter.

A.2.2.2 Pure simulation data generation

The pure simulation state and output predictions that can be generated are $\hat{\boldsymbol{x}}_{k|0}$, and $\hat{\boldsymbol{y}}_{k|0}$ corresponding to each time instant t_k in the input data set. The predictions are generated by the (extended) Kalman filter without updating.

B

Statistical tests and residual analysis tools

In this appendix an outline of the mathematical details of the statistical tests and residual analysis tools applied within the grey-box modelling cycle described in Chapter 2 is given. Some of the statistical tests are incorporated in **CTSM** (see Appendix A) and some have been implemented in MATLAB, whereas the residual analysis tools have all been implemented in MATLAB.

B.1 Statistical tests

The idea of the statistical tests applied within the grey-box modelling cycle is to make inferences about the parameters of continuous-discrete stochastic state space models. These tests are therefore based on the properties of the parameter estimates provided by **CTSM**, and as shown in Appendix A these estimates are asymptotically Gaussian with the following mean and covariance:

$$E\{\hat{\boldsymbol{\theta}}\} = \boldsymbol{\theta} \tag{B.1}$$

$$V\{\hat{\boldsymbol{\theta}}\} = \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}} \boldsymbol{R} \boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}} \tag{B.2}$$

where the covariance matrix $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}$ is approximated by the inverse of the Hessian evaluated at the minimum of the objective function. This covariance matrix can be decomposed into a diagonal matrix $\boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}}$ of the standard deviations of the individual parameter estimates and the corresponding correlation matrix $\boldsymbol{R}$.

B.1.1 Marginal tests

As shown in Appendix A the asymptotic Gaussianity property also allows marginal t-tests to be performed to test the hypothesis that a given parameter θ_j is insignificant (H_0: $\theta_j = 0$) against the alternative that it is not (H_1: $\theta_j \neq 0$), but this is actually just a special case of a more general test.

Indeed, marginal t-tests can be performed to test the more general hypothesis:

$$H_0: \quad \theta_j = \theta_j^0 \tag{B.3}$$

against the corresponding alternative:

$$H_1: \quad \theta_j \neq \theta_j^0 \tag{B.4}$$

i.e. to test whether a given parameter θ_j has a specific value θ_j^0 or not. The test quantity can be computed from the parameter estimate $\hat{\theta}_j$ and the standard deviation of the estimate $\sigma_{\hat{\theta}_j}$ in the following way:

$$z^t(\hat{\theta}_j) = \frac{\hat{\theta}_j - \theta_j^0}{\sigma_{\hat{\theta}_j}} \tag{B.5}$$

Under H_0 this quantity is asymptotically t-distributed with a number of degrees of freedom DF that equals the total number of observations minus the number of estimated parameters as shown in Appendix A, i.e.:

$$z^t(\hat{\theta}_j) \ \in \ t(\mathrm{DF}) \tag{B.6}$$

and the critical region for a test on significance level α is given as follows:

$$z^t(\hat{\theta}_j) < t(\mathrm{DF})_{\frac{\alpha}{2}} \ \vee \ z^t(\hat{\theta}_j) > t(\mathrm{DF})_{1-\frac{\alpha}{2}} \tag{B.7}$$

B.1.2 Simultaneous tests

Due to correlations between the individual parameter estimates, a series of marginal tests cannot be used to make inferences about several parameters simultaneously. Instead a test based on a statistic that takes correlations into account must be used. One such statistic, which is also based on the property of asymptotic Gaussianity, is Wald's W-statistic (Kotz and Johnson, 1985), which can be applied to test the following general hypothesis:

$$H_0: \quad \boldsymbol{g}(\boldsymbol{\theta}) = \mathbf{0} \tag{B.8}$$

against the corresponding alternative:

$$H_1: \quad \boldsymbol{g}(\boldsymbol{\theta}) \neq \mathbf{0} \tag{B.9}$$

i.e. to test whether the restriction given by the k-dimensional vector function $\boldsymbol{g}(\cdot)$ is satistied or not. The W-statistic can be computed in the following way:

$$W(\boldsymbol{g}(\hat{\boldsymbol{\theta}})) = (\boldsymbol{g}(\hat{\boldsymbol{\theta}}))^T \left(\boldsymbol{g}'(\hat{\boldsymbol{\theta}})\Sigma_{\hat{\boldsymbol{\theta}}}(\boldsymbol{g}'(\hat{\boldsymbol{\theta}}))^T\right)^{-1} \boldsymbol{g}(\hat{\boldsymbol{\theta}}) \tag{B.10}$$

where:

$$\boldsymbol{g}'(\hat{\boldsymbol{\theta}}) = \frac{\partial \boldsymbol{g}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \tag{B.11}$$

Under H_0 this quantity is asymptotically χ^2-distributed with a number of degrees of freedom k that equals the dimension of the restriction, i.e.:

$$W(\boldsymbol{g}(\hat{\boldsymbol{\theta}})) \in \chi^2(k) \tag{B.12}$$

and the critical region for a test on significance level α is given as follows:

$$W(\boldsymbol{g}(\hat{\boldsymbol{\theta}})) > \chi^2(k)_{1-\alpha} \tag{B.13}$$

As a very important special case, a test based on Wald's W-statistic can be used to test the hypothesis that a given subset of the parameters $\boldsymbol{\theta}_* \subset \boldsymbol{\theta}$ are simultaneously insignificant (H_0: $\boldsymbol{\theta}_* = \mathbf{0}$) against the alternative that they are not (H_1: $\boldsymbol{\theta}_* \neq \mathbf{0}$). In this case the W-statistic can be computed as follows:

$$W(\hat{\boldsymbol{\theta}}_*) = \hat{\boldsymbol{\theta}}_*^T \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}_*}^{-1} \hat{\boldsymbol{\theta}}_* \tag{B.14}$$

where $\hat{\boldsymbol{\theta}}_* \subset \hat{\boldsymbol{\theta}}$ is the subset of the parameter estimates subjected to the test and $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}_*}$ is the covariance matrix of these estimates. This covariance matrix can be computed from the full covariance matrix as follows:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}_*} = \boldsymbol{E} \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} \boldsymbol{E}^T \tag{B.15}$$

where $\boldsymbol{E}$ is an appropriate permutation matrix, which can be constructed from a unit matrix by eliminating the rows corresponding to parameter estimates not subjected to the test. This W-statistic can also be computed as follows:

$$W(\hat{\boldsymbol{\theta}}_*) = (\boldsymbol{z}^t(\hat{\boldsymbol{\theta}}_*))^T \boldsymbol{R}_*^{-1} \boldsymbol{z}^t(\hat{\boldsymbol{\theta}}_*) \tag{B.16}$$

where $\boldsymbol{z}^t(\hat{\boldsymbol{\theta}}_*)$ is a vector of marginal t-test quantities corresponding to the parameter estimates subjected to the test and $\boldsymbol{R}_*$ is the corresponding correlation matrix, which can be computed from the full correlation matrix as follows:

$$\boldsymbol{R}_* = \boldsymbol{E} \boldsymbol{R} \boldsymbol{E}^T \tag{B.17}$$

In either case the W-statistic corresponding to this special case is asymptotically χ^2-distributed under H_0 with $\dim(\hat{\boldsymbol{\theta}}_*)$ degrees of freedom.

B.2 Residual analysis tools

The idea of the residual analysis tools applied within the grey-box modelling cycle is to investigate the prediction capabilities of continuous-discrete stochastic state space models by examining residuals computed from validation data sets generated by **CTSM**, and, as shown in Appendix A, such data sets can be generated using either one-step-ahead prediction or pure simulation.

B.2.1 Standard tools

One of the most widely used methods for residual analysis is to compute and plot for an appropriate number of lags the standard correlation functions, i.e.:

- the *sample autocorrelation function* (SACF),
- the *sample partial autocorrelation function* (SPACF),
- and the *sample cross-correlation function* (SCCF),

which measure the correlation between current values of the residuals and lagged values of the residuals (SACF and SPACF) or the inputs (SCCF).

It must be noted that, although these tools are very well suited for investigating prediction capabilities, they can only be applied to stationary and equidistant time series of the residuals and inputs, unless proper precautions are taken.

B.2.1.1 Sample autocorrelation function

The sample autocorrelation function (SACF) of a stationary and equidistant time series $\{x_1, \ldots, x_N\}$ measures the correlation between current and lagged values of the underlying stochastic process $\{X_t\}$ and is defined as follows:

$$\hat{\rho}(k) = \frac{\hat{\gamma}(k)}{\hat{\gamma}(0)} \ , \ -N < k < N \tag{B.18}$$

where $\hat{\gamma}(\cdot)$ is the sample autocovariance function, which is defined as follows:

$$\hat{\gamma}(k) = \frac{1}{N} \sum_{t=1}^{N-k} (x_{t+k} - \overline{x})(x_t - \overline{x}) \ , \ 0 \leq k < N \tag{B.19}$$

$$\hat{\gamma}(k) = \hat{\gamma}(-k) \ , \ -N < k \leq 0 \tag{B.20}$$

where:

$$\overline{x} = \frac{1}{N} \sum_{t=1}^{N} x_t \tag{B.21}$$

The SACF $\hat{\rho}(\cdot)$ is an asymptotically unbiased estimate of the true autocorrelation function $\rho(\cdot)$ (Brockwell and Davis, 1991) and can therefore be used to perform marginal tests for all k of the following hypothesis:

$$H_0: \quad \rho(k) = 0 \tag{B.22}$$

against the corresponding alternative:

$$H_1: \quad \rho(k) \neq 0 \tag{B.23}$$

Under H_0 the test quantity $\hat{\rho}(k)$ is asymptotically $N(0, \frac{1}{N})$, and the critical region for a test on significance level α is given as follows:

$$\hat{\rho}(k) < N(0, \frac{1}{N})_{\frac{\alpha}{2}} \ \vee \ \hat{\rho}(k) > N(0, \frac{1}{N})_{1-\frac{\alpha}{2}} \tag{B.24}$$

This means that the test can easily be performed for a range of values of k simultaneously by plotting the SACF for the appropriate range and comparing with horizontal lines at the appropriate critical values. More complete details about the SACF are given by Brockwell and Davis (1991).

B.2.1.2 Sample partial autocorrelation function

The sample partial autocorrelation function (SPACF) of a stationary and equidistant time series $\{x_1, \ldots, x_N\}$ measures the correlation between current and lagged values of the underlying stochastic process $\{X_t\}$, adjusted for correlations with intermediate values, and is defined as follows:

$$\hat{\beta}(k) = \hat{\phi}_{kk} \ , \ 1 \leq k < N \tag{B.25}$$

where $\hat{\phi}_{kk}$ can be determined from values of the SACF as follows:

$$\begin{bmatrix} \hat{\rho}(0) & \hat{\rho}(1) & \cdots & \hat{\rho}(k-1) \\ \hat{\rho}(1) & \hat{\rho}(0) & \cdots & \hat{\rho}(k-2) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\rho}(k-1) & \hat{\rho}(k-2) & \cdots & \hat{\rho}(0) \end{bmatrix} \begin{bmatrix} \hat{\phi}_{k1} \\ \hat{\phi}_{k2} \\ \vdots \\ \hat{\phi}_{kk} \end{bmatrix} = \begin{bmatrix} \hat{\rho}(1) \\ \hat{\rho}(2) \\ \vdots \\ \hat{\rho}(k) \end{bmatrix} \ , \ k \geq 1 \tag{B.26}$$

The SPACF $\hat{\beta}(\cdot)$ is an asymptotically unbiased estimate of the true partial autocorrelation function $\beta(\cdot)$ (Brockwell and Davis, 1991) and can therefore be used to perform marginal tests for all k of the following hypothesis:

$$H_0\text{:} \quad \beta(k) = 0 \tag{B.27}$$

against the corresponding alternative:

$$H_1\text{:} \quad \beta(k) \neq 0 \tag{B.28}$$

Under H_0 the test quantity $\hat{\beta}(k)$ is again asymptotically $N(0, \frac{1}{N})$, and the critical region for a test on significance level α is given as follows:

$$\hat{\beta}(k) < N(0, \frac{1}{N})_{\frac{\alpha}{2}} \ \vee \ \hat{\beta}(k) > N(0, \frac{1}{N})_{1-\frac{\alpha}{2}} \tag{B.29}$$

Using a similar approach as the one described above for the SACF, this test can therefore easily be performed graphically for a range of values of k. More details about the SPACF are given by Brockwell and Davis (1991).

B.2.1.3 Sample cross-correlation function

The sample cross-correlation function (SCCF) between two stationary and equidistant time series $\{x_{i,1}, \ldots, x_{i,N}\}$ and $\{x_{j,1}, \ldots, x_{j,N}\}$ measures the correlation between current values of the underlying stochastic process $\{X_{i,t}\}$ and lagged values of the underlying stochastic process $\{X_{j,t}\}$ and is defined as follows:

$$\hat{\rho}_{ij}(k) = \frac{\hat{\gamma}_{ij}(k)}{\sqrt{\hat{\gamma}_{ii}(0)\hat{\gamma}_{jj}(0)}} \ , \ -N < k < N \tag{B.30}$$

where $\hat{\gamma}_{ij}(k)$ are elements of the multivariate sample autocovariance function:

$$\hat{\Gamma}(k) = \begin{cases} \frac{1}{N}\sum_{t=1}^{N-k}(\boldsymbol{x}_{t+k} - \overline{\boldsymbol{x}})(\boldsymbol{x}_t - \overline{\boldsymbol{x}})^T \ , \ 0 \leq k < N-1 \\ \\ \frac{1}{N}\sum_{t=-k+1}^{N}(\boldsymbol{x}_{t+k} - \overline{\boldsymbol{x}})(\boldsymbol{x}_t - \overline{\boldsymbol{x}})^T \ , \ -N+1 \leq k < 0 \end{cases} \tag{B.31}$$

where:

$$\boldsymbol{x}_t = [x_{i,t} \ x_{j,t}]^T \tag{B.32}$$

and:

$$\overline{\boldsymbol{x}} = \frac{1}{N}\sum_{t=1}^{N}\boldsymbol{x}_t \tag{B.33}$$

The SCCF $\hat{\rho}_{ij}(\cdot)$ is an asymptotically unbiased estimate of the true cross-correlation function $\rho_{ij}(\cdot)$ (Brockwell and Davis, 1991) and can therefore be used to perform marginal tests for all k of the following hypothesis:

$$H_0\text{:} \quad \rho_{ij}(k) = 0 \tag{B.34}$$

against the corresponding alternative:

$$H_1\text{:} \quad \rho_{ij}(k) \neq 0 \tag{B.35}$$

If either $\{X_{i,t}\}$ or $\{X_{j,t}\}$ is a white noise process or if *pre-whitening* of one or both of these processes is used (Brockwell and Davis, 1991), the test quantity $\hat{\rho}_{ij}(k)$ is again asymptotically $N(0, \frac{1}{N})$ under H_0, and the critical region for a test on significance level α is given as follows:

$$\hat{\rho}_{ij}(k) < N(0, \frac{1}{N})_{\frac{\alpha}{2}} \ \vee \ \hat{\rho}_{ij}(k) > N(0, \frac{1}{N})_{1-\frac{\alpha}{2}} \tag{B.36}$$

Using a similar approach as the one described above for the SACF, this test can therefore easily be performed graphically for a range of values of k. More details about the SCCF are given by Brockwell and Davis (1991).

B.2.2 Advanced tools

The standard tools for residual analysis measure the degree of linear dependency and therefore fail to detect certain nonlinear dependencies, but as shown by Nielsen and Madsen (2001a) generalized tools can be used instead, i.e.:

- the *lag dependence function* (LDF),
- the *partial lag dependence function* (PLDF),
- the *crossed lag dependence function* (CLDF),
- and the *nonlinear lag dependence function* (NLDF),

which are all based on the close relation between correlation coefficients and values of the coefficients of determination for regression models but extend from linear to nonlinear systems by incorporating nonparametric regression.

As well as for the standard tools, it must be noted that these tools can only be applied to stationary and equidistant time series of the residuals and inputs.

B.2.2.1 Lag dependence function

The lag dependence function (LDF), which is a generalization of the SACF, is based on the equivalence[1] between the squared correlation coefficient between the stochastic variables Y and X_k, which is defined as follows:

$$\rho^2_{0(k)} = \frac{V\{Y\} - V\{Y|X_k\}}{V\{Y\}} \tag{B.37}$$

and the coefficient of determination of a linear regression of a series of observations of Y on a series of observations of X_k, i.e.:

$$R^2_{0(k)} = \frac{SS_0 - SS_{0(k)}}{SS_0} \tag{B.38}$$

where $SS_{0(k)}$ is the sum of squares of the residuals from the regression and:

$$SS_0 = \sum_{i=1}^{N}(y_i - \frac{1}{N}\sum_{i=1}^{N} y_i)^2 \tag{B.39}$$

For a time series $\{x_1, \ldots, x_N\}$ of observations of a stationary process $\{X_t\}$, the squared SACF at lag k is equivalent to the squared correlation coefficient $\rho^2_{0(k)}$ between X_t and X_{t-k}, and it can therefore be closely approximated by the corresponding value of $R^2_{0(k)}$ obtained from a linear regression of observations of X_t on observations of X_{t-k}. By replacing the linear regression with a

[1] $R^2_{0(k)}$ is the ML estimate of $\rho^2_{0(k)}$ when Gaussianity is assumed.

nonparametric estimate of the conditional mean $f_k(x) = E\{X_t|X_{t-k} = x\}$, the LDF can be defined as a straightforward extension of the SACF as follows:

$$\mathrm{LDF}(k) = \mathrm{sign}(\hat{f}_k(b) - \hat{f}_k(a))\sqrt{\tilde{R}^2_{0(k)}} \ , \ 1 \leq k < N \tag{B.40}$$

where a and b are the minimum and maximum over the range of observations and $\tilde{R}^2_{0(k)}$ is the corresponding value of the coefficient of determination, i.e.:

$$\tilde{R}^2_{0(k)} = \frac{SS_0 - \widetilde{SS}_{0(k)}}{SS_0} \tag{B.41}$$

where $\widetilde{SS}_{0(k)}$ is the sum of squares of the appropriate residuals. The sign in the above definition is included to provide information about the average slope of the nonparametric estimate of the conditional mean. This estimate can be computed by using a nonparametric smoothing technique, e.g. basic kernel smoothing or locally-weighted regression (see Appendix C for details).

Being an extension of the SACF, the LDF can be interpreted as being, for each k, the part of the overall variation in the observations of X_t, which can be explained by the observations of X_{t-k}. Like the SACF, the LDF can therefore be used to perform tests of correlation for a range of values of k simultaneously by plotting the LDF for the appropriate range and comparing with appropriate confidence limits. These limits must be calculated by means of a *bootstrap* method (Nielsen and Madsen, 2001a) in this case, because they depend on the characteristics of the particular nonparametric smoothing technique used.

B.2.2.2 Partial lag dependence function

The partial lag dependence function (PLDF), which is a generalization of the SPACF, is based on the equivalence[2] between the squared partial correlation coefficient between the stochastic variable $(Y|X_1, \ldots, X_{k-1})$ and the stochastic variable $(X_k|X_1, \ldots, X_{k-1})$, which is defined as follows:

$$\rho^2_{(0k)|(1,\ldots,k-1)} = \frac{V\{Y|X_1, \ldots, X_{k-1}\} - V\{Y|X_1, \ldots, X_k\}}{V\{Y|X_1, \ldots, X_{k-1}\}} \tag{B.42}$$

and the following coefficient of determination:

$$R^2_{(0k)|(1,\ldots,k-1)} = \frac{SS_{0(1,\ldots,k-1)} - SS_{0(1,\ldots,k)}}{SS_{0(1,\ldots,k-1)}} \tag{B.43}$$

where $SS_{0(1,\ldots,k-1)}$ is the sum of squares of the residuals from a linear regression of a series of observations of Y on a series of observations of $(X_1, \ldots, X_{k-1})$ and $SS_{0(1,\ldots,k)}$ is the sum of squares of the residuals from a linear regression of a series of observations of Y on a series of observations of $(X_1, \ldots, X_k)$.

[2] $R^2_{(0k)|(1,\ldots,k-1)}$ is the ML estimate of $\rho^2_{(0k)|(1,\ldots,k-1)}$ when Gaussianity is assumed.

For a time series $\{x_1, \ldots, x_N\}$ of observations of a stationary process $\{X_t\}$, the squared SPACF at lag k is equivalent to the squared partial correlation coefficient $\rho^2_{(0k)|(1,\ldots,k-1)}$ between the stochastic variable $(X_t|X_{t-1}, \ldots, X_{t-(k-1)})$ and the stochastic variable $(X_{t-k}|X_{t-1}, \ldots, X_{t-(k-1)})$. It can therefore be closely approximated by the value of $R^2_{(0k)|(1,\ldots,k-1)}$ obtained from a linear regression of observations of X_t on observations of $(X_{t-1}, \ldots, X_{t-(k-1)})$ and a linear regression of observations of X_t on observations of $(X_{t-1}, \ldots, X_{t-k})$, i.e. by fitting the following set of auto-regressive models:

$$X_t = \phi_{j0} + \phi_{j1}X_{t-1} + \cdots + \phi_{jj}X_{t-j} + e_t \ , \ j = k-1, k \tag{B.44}$$

By replacing the set of auto-regressive models with a set of additive models:

$$X_t = f_{j0} + f_{j1}(X_{t-1}) + \cdots + f_{jj}(X_{t-j}) + e_t \ , \ j = k-1, k \tag{B.45}$$

where each f_{ji} is estimated nonparametrically (see Appendix C for details), the PLDF can be defined as a straightforward extension of the SPACF as follows:

$$\text{PLDF}(k) = \text{sign}(\hat{f}_{kk}(b) - \hat{f}_{kk}(a))\sqrt{\tilde{R}^2_{(0k)|(1,\ldots,k-1)}} \ , \ 1 \leq k < N \tag{B.46}$$

where a and b are again the minimum and maximum over the observations and $\tilde{R}^2_{(0k)|(1,\ldots,k-1)}$ is the corresponding coefficient of determination, i.e.:

$$\tilde{R}^2_{(0k)|(1,\ldots,k-1)} = \frac{\widetilde{SS}_{0(1,\ldots,k-1)} - \widetilde{SS}_{0(1,\ldots,k)}}{\widetilde{SS}_{0(1,\ldots,k-1)}} \tag{B.47}$$

where $\widetilde{SS}_{0(1,\ldots,k-1)}$ and $\widetilde{SS}_{0(1,\ldots,k)}$ are the sums of squares of the appropriate residuals. Again, the sign in the above definition is included to provide information about the average slope of the nonparametric estimates.

Being an extension of the SPACF, the PLDF can be interpreted as being, for each k, the relative decrease in one-step-ahead prediction variation when including X_{t-k} as an extra predictor. Thus, like the SPACF, the PLDF can be used to graphically perform tests of partial correlation for a range of values of k, using a similar approach as for the LDF to compute confidence limits.

B.2.2.3 Crossed lag dependence function

The crossed lag dependence function (CLDF), which is a generalization of the SCCF, is defined analogously to the LDF as follows:

$$\text{CLDF}(k) = \text{sign}(\hat{f}_k(b) - \hat{f}_k(a))\sqrt{\tilde{R}^2_{0(k)}} \ , \ 1 \leq k < N \tag{B.48}$$

where the estimate of the conditional mean $f_k(x) = E\{X_t|X_{t-k} = x\}$ is replaced with an estimate of the conditional mean $f_k(x) = E\{X_{i,t}|X_{j,t-k} = x\}$.

Being a generalization of the SCCF, the CLDF is a measure of the degree of dependency between current values of one time series and lagged values of another time series. Thus, like the SCCF, the CLDF can be used to graphically perform tests of cross-correlation for a range of values of k, using a similar approach as the one mentioned above for the LDF to compute confidence limits.

B.2.2.4 Nonlinear lag dependence function

The SACF is a measure of the degree of linear dependency between current and lagged values of a time series and the LDF is an extension, which measures the degree of both linear and nonlinear dependency. A measure of the degree of strictly nonlinear dependency is provided by the nonlinear lag dependence function (NLDF), which is defined analogously to the LDF as follows:

$$\mathrm{NLDF}(k) = \mathrm{sign}(\hat{f}_k(b) - \hat{f}_k(a))\sqrt{\tilde{R}^2_{0(k)}} \ , \ 1 \leq k < N \tag{B.49}$$

where the term SS_0 in the definition of $\tilde{R}^2_{0(k)}$ is replaced with the sum of squares $SS_{0(k)}$ of the residuals from a linear regression of a series of observations of X_t on a series of observations of X_{t-k}, i.e.:

$$\tilde{R}^2_{0(k)} = \frac{SS_{0(k)} - \widetilde{SS}_{0(k)}}{SS_{0(k)}} \tag{B.50}$$

The NLDF can be used to graphically perform tests of strictly nonlinear correlation for a range of values of k. A discussion of how to compute confidence limits for this type of test and more details about all of the lag dependence functions in general is given by Nielsen and Madsen (2001a).

C

Nonparametric methods

In this appendix an outline of the mathematical details of the nonparametric methods applied within the grey-box modelling cycle described in Chapter 2 is given. These methods, which have all been implemented in MATLAB, are applied for computing the lag dependence functions used for residual analysis (see Appendix B) and for nonparametric modelling of functional relations.

C.1 Kernel smoothing

The core nonparametric method is univariate kernel smoothing, which is a method that uses a training data set $(\boldsymbol{x}, \boldsymbol{y}) = \{x_k, y_k\}_{k=1}^{N}$ of observations of a predictor variable X and a response variable Y to compute a smoothed estimate of the response variable for a given value of the predictor variable.

More specifically, univariate kernel smoothing assumes the following relationship between the response variable and the predictor variable:

$$Y = f(X) + e \ , \ e \in N(0, \sigma^2) \tag{C.1}$$

and essentially uses the training data set to compute the conditional mean:

$$\hat{y}_0 = \hat{f}(x_0) = E\{Y|X = x_0\} \tag{C.2}$$

for a given value x_0 of the predictor variable. This section outlines some of the details of univariate kernel smoothing. More information can be found in the books of Hastie and Tibshirani (1990) and Hastie *et al.* (2001).

C.1.1 Basic kernel smoothing

The simplest kernel smoother is the *Nadaraya-Watson kernel weighted average*, which can be computed as follows for a single value x_0 of the predictor variable:

$$\hat{y}_0 = \frac{\sum_{k=1}^{N} K_\lambda(\frac{|x_k - x_0|}{\lambda}) y_k}{\sum_{k=1}^{N} K_\lambda(\frac{|x_k - x_0|}{\lambda})} \tag{C.3}$$

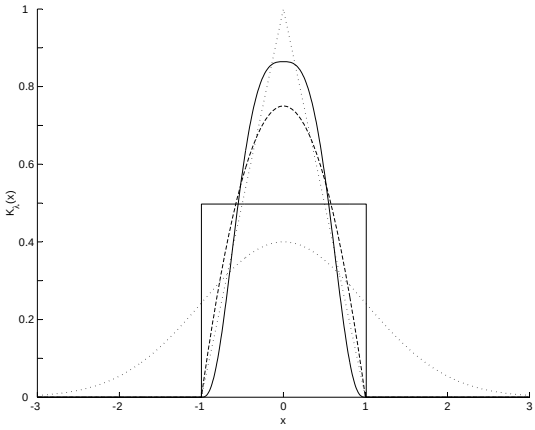


Figure C.1. Various kernel functions. Solid line: Box; dotted line: Triangular; Solid line: Tri-cube; dashed line: Epanechnikov; dotted line: Gaussian.

where $\hat{y}_0$ is the *fit* and K_λ is a *kernel* function with *bandwidth* λ. The kernel function is a symmetric weight function that assigns weights to observations close to x_0. Several such functions are available as shown in Figure C.1, e.g.:

- the *box* kernel:
$$K_\lambda(x) = \begin{cases} \frac{1}{2} & , \quad |x| \leq 1 \\ 0 & , \quad \text{otherwise} \end{cases} \tag{C.4}$$

- the *triangular* kernel:
$$K_\lambda(x) = \begin{cases} 1 - |x| & , \quad |x| \leq 1 \\ 0 & , \quad \text{otherwise} \end{cases} \tag{C.5}$$

- the *tri-cube* kernel:
$$K_\lambda(x) = \begin{cases} (1 - x^3)^3 & , \quad |x| \leq 1 \\ 0 & , \quad \text{otherwise} \end{cases} \tag{C.6}$$

- the *Epanechnikov* kernel:
$$K_\lambda(x) = \begin{cases} \frac{3}{4}(1 - x^2) & , \quad |x| \leq 1 \\ 0 & , \quad \text{otherwise} \end{cases} \tag{C.7}$$

- and the *Gaussian* kernel:
$$K_\lambda(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2}) \tag{C.8}$$

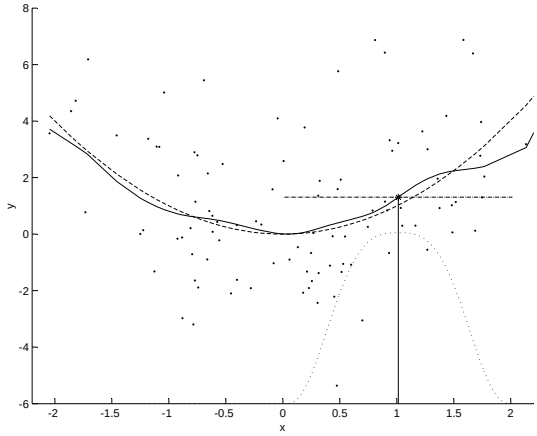


Figure C.2. Graphical illustration of how the Nadaraya-Watson kernel weighted average is computed. Solid vertical line: predictor value of interest; dotted curve: Tri-cube kernel with $\lambda = 1$ (re-scaled); dash-dotted horizontal line: Local average; asterisk: Local fit; solid curve: Overall fit; dashed curve: True curve.

The Gaussian kernel is the only one of these kernels that does not have *compact support*, which simply means that it is the only one that is unbounded.

In the more general case of a vector $\boldsymbol{x} = \{x_i\}_{i=1}^{N_{\text{fit}}}$ of values of the predictor variable, the kernel weighted average can be computed as follows:

$$\hat{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{y} \tag{C.9}$$

where $\hat{\boldsymbol{y}} = \{\hat{y}_i\}_{i=1}^{N_{\text{fit}}}$ is a vector of the corresponding fits and $\boldsymbol{S}$ is the *smoother matrix*, which is given by the following element entries:

$$\{s_{ij}\} = \frac{K_\lambda(\frac{|x_j - x_i|}{\lambda})}{\sum_{k=1}^{N} K_\lambda(\frac{|x_k - x_i|}{\lambda})} \ , \ i = 1, \ldots, N_{\text{fit}} \ , \ j = 1, \ldots, N \tag{C.10}$$

If the fit is computed for the exact predictor values in the training data set, the smoother matrix is a square matrix. Moreover, if a kernel that has compact support is used, the smoother matrix is often sparse, so to reduce the storage requirements for this matrix as well as the otherwise extensive computational load associated with the linear operation in (C.9), the specific implementation of basic kernel smoothing in MATLAB is based on a sparse matrix format.

Figure C.2 is a graphical illustration of how the Nadaraya-Watson kernel weighted average is computed: Kernel weights (dotted curve) are assigned to observations (dots) close to the predictor value of interest (indicated with a solid vertical line) to compute the local average over the range of the kernel function (dash-dotted horizontal line). Only the value at the particular predictor value

(indicated with an asterisk) is used, however, and by repeating the procedure for several predictor values a smoothed curve (solid curve) that approximates the true curve (dashed curve) can be constructed. More information about the basics of kernel smoothing is given by Hastie and Tibshirani (1990).

C.1.2 Locally-weighted regression

The Nadaraya-Watson kernel weighted average essentially fits a constant locally and as argued by Hastie *et al.* (2001) this approach may give severe bias, particularly on the boundaries of the range of predictor values in the training data set, but the bias can be removed by instead fitting a polynomial locally.

This approach is called *locally-weighted regression* and the corresponding fit can be computed as follows for a single value x_0 of the predictor variable:

$$\hat{y}_0 = \hat{\alpha}(x_0) + \sum_{j=1}^{d} \hat{\beta}_j(x_0) x_0^j \tag{C.11}$$

where $\hat{\alpha}(x_0)$ and $\hat{\beta}_j(x_0)$, $j = 1, \ldots, d$, are computed by solving the following locally-weighted d'th order polynomial regression problem:

$$\min_{\alpha,\beta_1,\ldots,\beta_d} \sum_{k=1}^{N} K_\lambda(\frac{x_k - x_0}{\lambda}) \left(y_k - \alpha + \sum_{j=1}^{d} \beta_j x_k^j \right)^2 \tag{C.12}$$

In the more general case of a vector $\boldsymbol{x} = \{x_i\}_{i=1}^{N_{\text{fit}}}$ of values of the predictor variable the locally-weighted regression fit can be computed as follows:

$$\hat{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{y} \tag{C.13}$$

where $\hat{\boldsymbol{y}} = \{\hat{y}_i\}_{i=1}^{N_{\text{fit}}}$ is a vector of the corresponding fits and $\boldsymbol{S}$ is the smoother matrix, which is given by the following row entries:

$$\{s_i\} = \begin{bmatrix} 1 & x_i & \cdots & x_i^d \end{bmatrix} \left(\boldsymbol{B}^T \boldsymbol{W}(x_i) \boldsymbol{B} \right)^{-1} \boldsymbol{B}^T \boldsymbol{W}(x_i) \ , \ i = 1, \ldots, N_{\text{fit}} \tag{C.14}$$

where:

$$\boldsymbol{B} = \begin{bmatrix} 1 & x_1 & \cdots & x_1^d \\ 1 & x_2 & \cdots & x_2^d \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & \cdots & x_N^d \end{bmatrix} \tag{C.15}$$

and:

$$\boldsymbol{W}(x_i) = \begin{bmatrix} K_\lambda(\frac{|x_1 - x_i|}{\lambda}) & 0 & \cdots & 0 \\ 0 & K_\lambda(\frac{|x_2 - x_i|}{\lambda}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_\lambda(\frac{|x_N - x_i|}{\lambda}) \end{bmatrix} \tag{C.16}$$

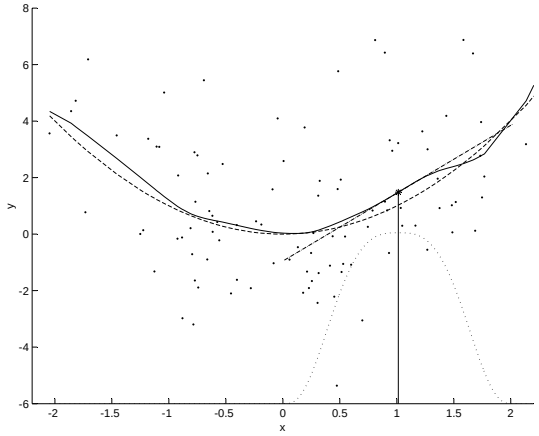


Figure C.3. Graphical illustration of how the locally-weighted linear regression fit is computed. Solid vertical line: predictor value of interest; dotted curve: Tri-cube kernel with $\lambda = 1$ (re-scaled); dash-dotted line: Local linear regression result; asterisk: Local fit; solid curve: Overall fit; dashed curve: True curve.

If the fit is computed for the exact predictor values in the training data set, the smoother matrix is also a square matrix in this case. Likewise, the smoother matrix is also often sparse in this case, so to reduce the storage requirements for this matrix as well as the otherwise extensive computational load associated with the linear operation in (C.13), the specific implementation of locally-weighted regression in MATLAB is also based on a sparse matrix format.

Figure C.3 is a graphical illustration of how the locally-weighted regression fit is computed in the linear case: Kernel weights (dotted curve) are assigned to observations (dots) close to the predictor value of interest (indicated with a solid vertical line) to compute the local linear regression result over the range of the kernel function (dash-dotted line). Again, only the value at the particular predictor value (indicated with an asterisk) is used, and by repeating the procedure for several predictor values a smoothed curve (solid curve) that approximates the true curve (dashed curve) can be constructed. More information about locally-weighted regression is given by Hastie *et al.* (2001).

C.1.3 Bandwidth issues

In order to apply locally-weighted regression, three choices must be made. The kernel function must be selected, the order of the local polynomial must be chosen, and the bandwidth of the kernel must be determined. The kernel function and the order of the local polynomial usually have much less impact on the resulting fit than the bandwidth. This is due to the fundamental bias-

variance trade-off in kernel smoothing (Hastie and Tibshirani, 1990), which means that small bandwidths give small bias but large variance whereas large bandwidths, on the other hand, give small variance but large bias.

Actually, determining the bandwidth of the kernel is a two-step procedure, because it involves a choice of the type of bandwidth as well as its size. There are two different types of kernel bandwidths: Metric bandwidths and nearest neighbour bandwidths. *Metric bandwidths* are fixed-size bandwidths, which remain constant over the entire range of predictor values in the training data set, i.e. $\lambda = c$, where c is a constant. *Nearest neighbour bandwidths*, on the other hand, are variable-size bandwidths, which adapt to the local density of the predictor values in the training data set by adjusting to encompass a fixed number K of nearest neighbours to the predictor value of interest, i.e. $\lambda = |x_K - x_0|$, where x_K is the K'th closest x_k to x_0. Because of this construction, nearest neighbour bandwidths can only be used with kernel functions that have compact support.

Given a particular type of bandwidth, its size, i.e. c or K, must somehow be determined to give a result that trades off bias and variance in an appropriate way. Hastie and Tibshirani (1990) discuss this extensively and suggest one of the following approaches: Calibration based on the effective degrees of freedom of the smoother, or optimisation based on an estimate of the prediction error.

The effective degrees of freedom of a smoother is defined as the trace of the smoother matrix when computing the fit for all predictor values in the training data set, i.e. $\mathrm{tr}(\boldsymbol{S})$, and this quantity can be used to determine the bandwidth by iteratively calibrating the amount of smoothing. This interactive approach is not necessarily optimal, so automatic bandwidth optimisation is preferred.

Automatic bandwidth optimisation seeks to find the bandwidth that minimizes a given estimate of the prediction error. A number of such estimates are discussed by Hastie *et al.* (2001). The simplest and most generally applicable of these is the so-called *cross-validation* (CV) statistic, which may be defined in one of two different ways, depending on the data available for validation.

If a separate validation data set $(\boldsymbol{x}_{\mathrm{val}}, \boldsymbol{y}_{\mathrm{val}}) = \{x_{\mathrm{val},i}, y_{\mathrm{val},i}\}_{i=1}^{N_{\mathrm{val}}}$ of corresponding values of the predictor variable and the response variable is available, the optimal size of the bandwidth can be determined in the following way:

$$\hat{\lambda} = \arg\min_{\lambda} \mathrm{CV}(\lambda) = \arg\min_{\lambda} \frac{1}{N_{\mathrm{val}}} \sum_{i=1}^{N_{\mathrm{val}}} (y_{\mathrm{val},i} - \hat{y}_i)^2 \tag{C.17}$$

where $\hat{y}_i$ is the fit for $x_{\mathrm{val},i}$ computed with a given value of λ from the training data set. This formulation can be used for metric bandwidths, i.e. to determine c, as well as for nearest neighbour bandwidths, i.e. to determine K.

If a separate validation data set is not available the optimal size of the bandwidth can be determined by means of *k-fold cross-validation* on the training data set. With this method the training data set is divided into k distinct groups (e.g. by assigning every k'th observation in the sorted data set to the

same group to obtain similar densities in all groups), the fit is computed for each predictor value in the first group using the data from the other groups, the corresponding contribution to the cross-validation statistic is computed and the procedure is repeated for all groups. This way the entire training data set is used for validation as well as for training, but overfitting is avoided. To formalize this, the optimal size of the bandwidth can be determined by k-fold cross-validation on the training data set in the following way:

$$\hat{\lambda} = \arg\min_{\lambda} \mathrm{CV}(\lambda) = \arg\min_{\lambda} \frac{1}{N} \sum_{i=1}^{N} \left(y_i - \hat{y}_i^{-\kappa(i)} \right)^2 \tag{C.18}$$

where $\hat{y}_i^{-\kappa(i)}$ is the fit for x_i computed with a given value of λ from the training data set without the observations indexed by the function $\kappa(i)$, which returns the indices of all observations in the group containing (x_i, y_i). This formulation can also be used for metric as well as for nearest neighbour bandwidths.

Kernel smoothers are linear smoothers, which means that the fit can be computed through a linear operation as in (C.9) or (C.13) by means of the smoother matrix, which, if the fit is computed for all predictor values in the training data set, is a square matrix. This can be utilized to derive a closed-form version of the above k-fold cross-validation statistic, which can be computed from a single fit on the entire training data set, and which in turn allows the optimal size of the bandwidth to be determined in the following way:

$$\hat{\lambda} = \arg\min_{\lambda} \mathrm{CV}(\lambda) = \arg\min_{\lambda} \frac{1}{N} \sum_{i=1}^{N} \left(\frac{y_i + \sum_{j \in \kappa(i)} s_{ij}(y_j - y_i) - \hat{y}_i}{1 - \sum_{j \in \kappa(i)} s_{ij}} \right)^2 \tag{C.19}$$

where $\hat{y}_i$ is the fit for x_i computed with a given value of λ from the entire training data set, and where s_{ij} is an element of the corresponding smoother matrix. Strictly speaking, this closed-form formulation can only be used for metric bandwidths, because of the fact that, for nearest neighbour bandwidths, the removal of one or more points implied by the formulation affects the local density of the predictor values to which such bandwidths adapt.

C.1.4 Confidence intervals

To provide an assessment of the uncertainty of a kernel smoother, approximate confidence intervals can be computed by means of the *nonparametric bootstrap* as discussed by Hastie *et al.* (2001). The idea of this method is to create a number of new data sets of the same size as the original training data set by randomly drawing (with replacement) observations from the training data set and then compute the fit for all of the new data sets and use the information gathered from this to construct pointwise confidence intervals.

More specifically, if B new data sets or *bootstrap samples* are created, the same kernel smoother as was applied to compute the nominal fit on the original

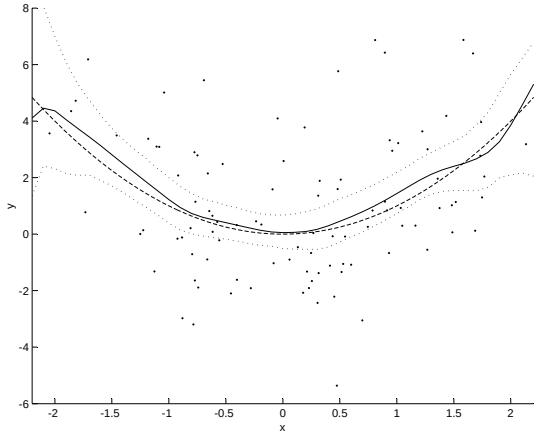


Figure C.4. Example of a locally-weighted linear regression fit (tri-cube kernel with optimal nearest neighbour bandwidth determined using 5-fold cross-validation) with 95% confidence limits computed from 1000 bootstrap replicates. Solid curve: Nominal fit; dotted curves: 95% confidence limits; dashed curve: True curve.

training data set is applied to each of the bootstrap samples in turn to produce a total of B *bootstrap replicates* of the fit for all predictor values of interest, whereupon the particular replicates corresponding to the appropriate percentiles of the total set of replicates (2.5 and 97.5 for 95% confidence intervals) are found and plotted along with the nominal fit as shown in Figure C.4.

C.2 Additive models

Another important nonparametric method is additive model fitting, which is a method that uses a training data set $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p, \boldsymbol{y}) = \{x_{1k}, \ldots, x_{pk}, y_k\}_{k=1}^{N}$ of observations of several predictor variables $X_1, \ldots, X_p$ and a single response variable Y to compute a smoothed estimate of the response variable for a given set of values of the predictor variables. In other words, additive model fitting is a multivariate nonparametric smoothing method. Several such methods are available (Hastie *et al.*, 2001), but additive model fitting has the particular advantage that it circumvents the *curse of dimensionality*, which tends to render such methods infeasible in higher dimensions (Hastie *et al.*, 2001).

More specifically, additive model fitting assumes the following relationship between the response variable Y and the predictor variables $X_1 \ldots, X_p$:

$$Y = \alpha + \sum_{j=1}^{p} f_j(X_j) + e \ , \ e \in N(0, \sigma^2) \tag{C.20}$$

and essentially uses the training data set to compute the conditional mean:

$$\hat{y}_0 = \hat{\alpha} + \sum_{j=1}^{p} \hat{f}_j(x_{j0}) = E\{Y|X_1 = x_{10}, \ldots, X_p = x_{p0}\} \tag{C.21}$$

for a given set of values $x_{10}, \ldots, x_{p0}$ of the predictor variables, which can be done by applying the so-called *backfitting algorithm*. This section outlines the details of this algorithm and discusses a number of other important aspects of additive model fitting. More information about this topic can be found in the books of Hastie and Tibshirani (1990) and Hastie *et al.* (2001).

C.2.1 The backfitting algorithm

The idea of the backfitting algorithm for fitting additive models is to compute the constant $\hat{\alpha}$ and then recursively adjust each of the predictor variable contributions $\hat{f}_j(x_{j0})$ one at a time until they remain unchanged.

The constant $\hat{\alpha}$ is computed as the average of the values of the response variable in the training data set and the predictor variable contributions $\hat{f}_j(x_{j0})$ are computed by repeatedly applying a univariate nonparametric smoother to residuals computed by subtracting the constant $\hat{\alpha}$ and the other predictor variable contributions from the values of the response variable. Applying univariate kernel smoothers to fit each of the predictor variable contributions and assuming that the overall fit is to be computed for the exact sets of predictor values in the training data set, this can be formalized as follows:

1. Set $\hat{\alpha} = \frac{1}{N}\sum_{k=1}^{N} y_k$ and initialize $\hat{\boldsymbol{f}}_j(\boldsymbol{x}_j) = \boldsymbol{0}$, $j = 1, \ldots, p$.

2. Compute for $j = 1, \ldots, p$:

$$\hat{\boldsymbol{f}}_j(\boldsymbol{x}_j) = \boldsymbol{S}_j \left(\boldsymbol{y} - \hat{\alpha} - \sum_{i \neq j} \hat{\boldsymbol{f}}_i(\boldsymbol{x}_i) \right) \tag{C.22}$$

$$\hat{\boldsymbol{f}}_j(\boldsymbol{x}_j) \leftarrow \hat{\boldsymbol{f}}_j(\boldsymbol{x}_j) - \frac{1}{N}\sum_{k=1}^{N} \hat{f}_j(x_{jk}) \tag{C.23}$$

3. Repeat 2 until $\hat{\boldsymbol{f}}_j(\boldsymbol{x}_j)$, $j = 1, \ldots, p$, change less than a given threshold.

4. Compute the overall fit:

$$\hat{\boldsymbol{y}} = \hat{\alpha} + \sum_{j=1}^{p} \hat{\boldsymbol{f}}_j(\boldsymbol{x}_j) \tag{C.24}$$

where $\hat{\boldsymbol{f}}_j(\boldsymbol{x}_j) = \{\hat{f}_j(x_{jk})\}_{k=1}^N$ is a vector of the contributions from the predictor variable x_j to the resulting overall fit $\hat{\boldsymbol{y}} = \{\hat{y}_k\}_{k=1}^N$ and $\boldsymbol{S}_j$ is the corresponding smoother matrix, which can be computed by applying one of the kernel smoothers discussed in Section C.1.1 and Section C.1.2. The correction in (C.23), which forces the individual contributions to average zero over the training data set, is introduced to prevent the lack of convergence of the backfitting algorithm that may otherwise result (Hastie *et al.*, 2001).

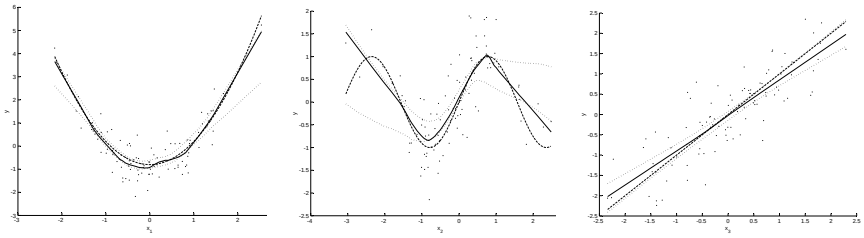
C.2.2 Bandwidth issues

Relying on kernel smoothers to fit the contributions from the individual predictor variables and thus requiring appropriate selection of a set of bandwidths, the bandwidth issues discussed for kernel smoothers in Section C.1.3 are also important for additive model fitting. In principle, the same arguments apply with respect to the importance of appropriately choosing both the type and the size of the bandwidths, but the problem is a bit more complex in this case, because of the multivariate nature of additive models, especially with respect to using automatic bandwidth optimisation. Ideally, all bandwidths should be determined simultaneously by solving a multiple bandwidth optimisation problem based on an appropriate cross-validation statistic in an outer loop surrounding the entire backfitting algorithm, but this approach renders additive model fitting extremely slow. Alternatively, the individual bandwidths could be determined separately by solving a set of single bandwidth optimisation problems based on k-fold cross-validation on the training data set in each backfitting iteration, but this approach also renders additive model fitting very slow, especially if a substantial number of backfitting iterations are needed.

A more feasible alternative is to determine the individual bandwidths separately by solving a set of such single bandwidth optimisation problems once and for all in the first backfitting iteration. This approach may seem very crude, but in fact the results obtained are only slightly different from the results obtained using bandwidth optimisation in all backfitting iterations.

C.2.3 Confidence intervals

The nonparametric bootstrap discussed for kernel smoothers in Section C.1.4 can also be applied to provide an assessment of the uncertainty of an additive model fit in the form of approximate confidence intervals. Ideally a number of bootstrap samples of the same size as the training data set should be drawn, the backfitting algorithm should be re-applied to all of these using the same kernel smoothers for the individual predictor variables as were applied to compute the nominal fit, and the information gathered from this should be used to construct pointwise confidence intervals. This approach is very slow, however, especially if many backfitting iterations are needed. A much faster alternative



(a) $\hat{f}_1(x_1)$. (b) $\hat{f}_2(x_2)$. (c) $\hat{f}_3(x_3)$.

Figure C.5. Example of an additive model fit on three predictor variables using locally-weighted linear regression (tri-cube kernels with optimal nearest neighbour bandwidths determined using 5-fold cross-validation) with 95% confidence limits computed from 1000 bootstrap replicates. Solid curves: Nominal fits; dotted curves: 95% confidence limits; dashed curves: True curves.

is to wait until the backfitting algorithm has converged and then apply the non-parametric bootstrap to each of the kernel smoothers used for the individual predictor variables. In other words separate sets of bootstrap samples are created for each predictor variable from the appropriate backfitting residuals, and the same kernel smoothers as were applied to compute the nominal fits in the last backfitting iteration are applied to produce separate sets of bootstrap replicates and the particular replicates corresponding to the appropriate percentiles are found and plotted along with the nominal fits as shown in Figure C.5.

D

Paper no. 1

The paper[1] included in this appendix is related to the parameter estimation element of the grey-box modelling cycle described in Chapter 2 and focuses on methods for parameter estimation in continuous-discrete stochastic state space models in general. The paper contains a condensed outline of the algorithms of **CTSM** as well as a comparison between this program and a program by Bohlin and Graebe (1995) and Bohlin (2001) implementing a similar estimation method. This comparison reveals some important differences between the two methods, which render the program by Bohlin and Graebe (1995) and Bohlin (2001) inappropriate for estimation of the parameters of the diffusion term and hence for application within the proposed grey-box modelling framework.

[1] The paper has been submitted for publication in *Automatica*.

Parameter Estimation in Stochastic Grey-Box Models

Niels Rode Kristensen[a], Henrik Madsen[b], Sten Bay Jørgensen[a]

[a]Department of Chemical Engineering, Technical University of Denmark, Building 229, DK-2800 Lyngby, Denmark

[b]Informatics and Mathematical Modelling, Technical University of Denmark, Building 321, DK-2800 Lyngby, Denmark

Abstract

An efficient and flexible parameter estimation scheme for grey-box models in the sense of systems of nonlinear discretely, partially observed Itô stochastic differential equations with measurement noise is presented along with a corresponding software implementation. The estimation scheme is based on the extended Kalman filter and features *maximum likelihood* as well as *maximum a posteriori* estimation on multiple independent data sets, including irregularly sampled data sets and data sets with occasional outliers and missing observations. The software implementation is compared to an existing software tool and proves to have superior estimation performance both in terms of quality of estimates and in terms of reproducibility. In particular, the new tool provides more accurate and consistent estimates of the parameters of the diffusion term.

Keywords: Grey-box models; parameter estimation; stochastic differential equations; maximum likelihood estimation; extended Kalman filter; estimation with missing observations; robust estimation; estimation accuracy; software.

D.1 Introduction

The development of various methods for advanced model-based control (Clarke *et al.*, 1987a,b; Bitmead *et al.*, 1990; Muske and Rawlings, 1993; Allgöwer and Zheng, 2000) and recent advances in sensor technology allowing these methods to be applied to an increasing number of complex physical, chemical and biological systems has rendered the development of high quality models for such systems very important. In particular, since a model must be able to predict the future evolution of the system to be controlled, it must capture the inherently nonlinear behaviour of many such systems and it must provide means to accommodate noise in the form of process noise due to approximation errors or unmodelled inputs and measurement noise due to imperfect measurements.

White-box models, derived from first principles, are often able to satisfy the former requirement but fail to satisfy the latter, whereas *black-box* models, developed with methods for system identification (Ljung, 1987; Söderström and Stoica, 1989), satisfy the latter but often fail to satisfy the former. Stochastic state space models or *grey-box* models, which consist of a set of stochastic differential equations (SDE's) describing the dynamics of the system in continuous time and a set of discrete time measurement equations, provide a way of combining the advantages of both model types by allowing prior physical knowledge to be incorporated and statistical methods for parameter estimation to be applied. Bohlin and Graebe (1995) even argue that such models provide a natural framework for modelling dynamic systems. Apart from the work by Bohlin and Graebe (1995) and earlier work by some of the authors of the present paper, mathematical modelling of dynamic systems based on SDE's has received limited attention in the control and system identification communities since Jazwinski (1970) and Åström (1970). This is evident from a series of review papers on the state of the art of identification of continuous time models (Young, 1981; Unbehauen and Rao, 1990, 1998). However, owing to the many potential benefits of grey-box models, it is the opinion of the authors of the present paper that the topic deserves much more attention.

Particular benefits of grey-box models as opposed to black-box models include the fact that physical knowledge and other prior information can be incorporated directly. This typically yields models with fewer and physically meaningful parameters, which are valid over much wider ranges of state space. As opposed to white-box models parameter estimation in grey-box models tends to give more reproducible results and less bias, because random effects due to process and measurement noise are not absorbed into the parameter estimates but specifically accounted for by the diffusion term and the measurement noise term. Furthermore, simultaneous estimation of the parameters of these terms as well in turn provides an estimate of the uncertainty of the model, upon which further model development can be based. In particular, estimates of the parameters of the diffusion term can be used to assess the quality of a model (Kristensen *et al.*, 2001), to discriminate between different models (Kristensen

et al., 2002a), and to pinpoint model deficiencies and subsequently uncover their structural origin (Kristensen *et al.*, 2002c). Thus, obtaining accurate and consistent estimates of the parameters of the diffusion term is very important.

The focus of the present paper is on estimation of unknown parameters in grey-box models in general, and the primary aim of the paper is to present an efficient and flexible scheme for performing the estimation and a software implementation of this scheme. A similar parameter estimation scheme and software tool has been presented by Bohlin and Graebe (1995), and a secondary aim of the paper is to outline how the two schemes differ and to demonstrate how these differences influence estimation performance. An important result is that the new tool provides more accurate and consistent estimates of the parameters of the diffusion term. The remainder of the paper is organized as follows: The mathematical basis of the estimation scheme is presented in Section D.2 and the software implementation is described in Section D.3. The differences between the estimation scheme presented here and the one by Bohlin and Graebe (1995) are outlined in Section D.4, where the influence on estimation performance is also demonstrated by means of simulation results. These results are discussed in Section D.5 and the conclusions of the paper are given in Section D.6.

D.2 Mathematical basis

This section contains a condensed outline of the mathematics behind the proposed parameter estimation scheme and of the algorithms of the corresponding software implementation (see Section D.3). A complete outline of the proposed estimation scheme is given by Kristensen *et al.* (2002d).

D.2.1 General model structure

Adapting the terminology of Bohlin and Graebe (1995), the term *grey-box model* will be used throughout this paper as an acronym for a model consisting of nonlinear discretely partially observed SDE's with measurement noise, i.e.:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{D.1}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{D.2}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathcal{X} \subset \mathbb{R}^n$ is a vector of state variables, $\boldsymbol{u}_t \in \mathcal{U} \subset \mathbb{R}^m$ is a vector of input variables, $\boldsymbol{y}_k \in \mathcal{Y} \subset \mathbb{R}^l$ is a vector of output variables, $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of parameters, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ and $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ are nonlinear functions, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

Remark 1. SDE's may be interpreted in the sense of either Stratonovich or Itô, but since the Stratonovich interpretation is less suitable for parameter estimation (Jazwinski, 1970; Åström, 1970), the Itô interpretation is adapted.

Remark 2. The diffusion term in (D.1) is assumed to be independent of the state variables, because this renders parameter estimation more feasible, but, as shown by Nielsen and Madsen (2001b), a transformation may be applied for a restricted class of systems with such dependences or *level effects*, allowing application of the proposed estimation scheme to such systems as well.

D.2.2 Parameter estimation methods

D.2.2.1 Maximum likelihood estimation

Given the model structure in (D.1)-(D.2) *maximum likelihood* (ML) estimates of the unknown parameters can be determined by finding the parameters $\boldsymbol{\theta}$ that maximize the likelihood function of a given sequence of measurements $\boldsymbol{y}_0$, $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k, \ldots, \boldsymbol{y}_N$. By introducing the notation:

$$\mathcal{Y}_k = [\boldsymbol{y}_k, \boldsymbol{y}_{k-1}, \ldots, \boldsymbol{y}_1, \boldsymbol{y}_0] \tag{D.3}$$

the likelihood function is the joint probability density:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = p(\mathcal{Y}_N|\boldsymbol{\theta}) \tag{D.4}$$

or equivalently:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} p(\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}) \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{D.5}$$

where the rule $P(A \cap B) = P(A|B)P(B)$ has been applied to form a product of conditional densities. In order to obtain an exact evaluation of the likelihood function, a general nonlinear filtering problem must be solved. Thus the initial probability density $p(\boldsymbol{y}_0|\boldsymbol{\theta})$ must be known and all subsequent conditional densities must be determined by successively solving Kolmogorov's forward equation and applying Bayes' rule (Jazwinski, 1970). In practice, this approach is computationally infeasible, however, and an alternative is needed. Nielsen *et al.* (2000a) have recently reviewed the state of the art with respect to parameter estimation in discretely observed Itô SDE's and in the general case of higher-order partially observed systems with measurement noise they conclude that only methods based on approximate nonlinear filters provide a computationally feasible solution to the problem. However, since the diffusion term in (D.1) has been assumed to be independent of the state variables, a simpler alternative can be used. More specifically, since the SDE's in (D.1) are driven by a Wiener process, and since increments of a Wiener process are Gaussian, it is reasonable to assume, under some regularity conditions, that the conditional densities can be well approximated by Gaussian densities, which means that a method based on the extended Kalman filter (EKF) can be applied. The assumption can (and should) be checked subsequent to the estimation (Holst

et al., 1992; Bak *et al.*, 1999). The Gaussian density is completely characterized by its mean and covariance, so by introducing the notation:

$$\hat{\boldsymbol{y}}_{k|k-1} = E\{\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}\} \tag{D.6}$$

$$\boldsymbol{R}_{k|k-1} = V\{\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}\} \tag{D.7}$$

and:

$$\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{D.8}$$

the likelihood function becomes:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1} \boldsymbol{\epsilon}_k\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{D.9}$$

and the parameter estimates can be determined by further conditioning on $\boldsymbol{y}_0$ and solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \{-\ln\left(L(\boldsymbol{\theta}; \mathcal{Y}_N|\boldsymbol{y}_0)\right)\} \tag{D.10}$$

For each set of parameters $\boldsymbol{\theta}$ in the optimisation, the innovations $\boldsymbol{\epsilon}_k$ and their covariances $\boldsymbol{R}_{k|k-1}$ are computed recursively by means of the EKF, which consists of the output *prediction* equations:

$$\hat{\boldsymbol{y}}_{k|k-1} = \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \tag{D.11}$$

$$\boldsymbol{R}_{k|k-1} = \boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^T + \boldsymbol{S} \tag{D.12}$$

the *innovation* equation:

$$\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{D.13}$$

the Kalman *gain* equation:

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{R}_{k|k-1}^{-1} \tag{D.14}$$

the *updating* equations:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\boldsymbol{\epsilon}_k \tag{D.15}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{R}_{k|k-1}\boldsymbol{K}_k^T \tag{D.16}$$

and the state *prediction* equations:

$$\frac{d\hat{\boldsymbol{x}}_{t|k}}{dt} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{t|k}, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \ , \ t \in [t_k, t_{k+1}[\tag{D.17}$$

$$\frac{d\boldsymbol{P}_{t|k}}{dt} = \boldsymbol{A}\boldsymbol{P}_{t|k} + \boldsymbol{P}_{t|k}\boldsymbol{A}^T + \boldsymbol{\sigma}\boldsymbol{\sigma}^T \ , \ t \in [t_k, t_{k+1}[\tag{D.18}$$

In the above equations the following shorthand notation has been applied:

$$\begin{aligned} \boldsymbol{A} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_t}|_{\hat{\boldsymbol{x}}_{k|k-1},\boldsymbol{u}_k,t_k} \ , \ \boldsymbol{C} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_t}|_{\hat{\boldsymbol{x}}_{k|k-1},\boldsymbol{u}_k,t_k} \\ \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_k,t_k,\boldsymbol{\theta}) \ , \ \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{u}_k,t_k,\boldsymbol{\theta}) \end{aligned} \tag{D.19}$$

Initial conditions for the EKF are $\hat{\boldsymbol{x}}_{t|t_0} = \boldsymbol{x}_0$ and $\boldsymbol{P}_{t|t_0} = \boldsymbol{P}_0$, which can either be pre-specified or estimated as a part of the overall problem. Being a linear filter, the EKF is sensitive to nonlinear effects, and the approximate solution obtained by solving (D.17)-(D.18) may be too crude (Jazwinski, 1970). Moreover, the assumption of Gaussian conditional densities is only likely to hold for small sample times (and should thus be checked subsequent to the estimation (Holst *et al.*, 1992; Bak *et al.*, 1999)). To provide a better approximation, the time interval $[t_k, t_{k+1}[$ is therefore subsampled, i.e. $[t_k, \ldots, t_j, \ldots, t_{k+1}[$, and the equations are linearized at each subsampling instant. This also means that direct numerical solution of (D.17)-(D.18) can be avoided by applying the analytical solutions to the corresponding linearized propagation equations:

$$\frac{d\hat{\boldsymbol{x}}_{t|j}}{dt} = \boldsymbol{f}_0 + \boldsymbol{A}(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{B}(\boldsymbol{u}_t - \boldsymbol{u}_j) \ , \ t \in [t_j, t_{j+1}[\tag{D.20}$$

$$\frac{d\boldsymbol{P}_{t|j}}{dt} = \boldsymbol{A}\boldsymbol{P}_{t|j} + \boldsymbol{P}_{t|j}\boldsymbol{A}^T + \boldsymbol{\sigma}\boldsymbol{\sigma}^T \ , \ t \in [t_j, t_{j+1}[\tag{D.21}$$

where the following shorthand notation has been applied:

$$\begin{aligned} \boldsymbol{A} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_t}|_{\hat{\boldsymbol{x}}_{j|j-1},\boldsymbol{u}_j,t_j} \ , \ \boldsymbol{B} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}_t}|_{\hat{\boldsymbol{x}}_{j|j-1},\boldsymbol{u}_j,t_j} \\ \boldsymbol{f}_0 = \boldsymbol{f}(\hat{\boldsymbol{x}}_{j|j-1},\boldsymbol{u}_j,t_j,\boldsymbol{\theta}) \ , \ \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_j,t_j,\boldsymbol{\theta}) \end{aligned} \tag{D.22}$$

The analytical solutions to (D.20)-(D.21) are:

$$\hat{\boldsymbol{x}}_{j+1|j} = \hat{\boldsymbol{x}}_{j|j} + \boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right)\boldsymbol{f}_0 + \left(\boldsymbol{A}^{-1}\left(\boldsymbol{\Phi}_s - \boldsymbol{I}\right) - \boldsymbol{I}\tau_s\right)\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{\alpha} \tag{D.23}$$

$$\boldsymbol{P}_{j+1|j} = \boldsymbol{\Phi}_s\boldsymbol{P}_{j|j}\boldsymbol{\Phi}_s^T + \int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T e^{\boldsymbol{A}^T s} ds \tag{D.24}$$

where $\tau_s = t_{j+1} - t_j$ and $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$, and where:

$$\boldsymbol{\alpha} = \frac{\boldsymbol{u}_{j+1} - \boldsymbol{u}_j}{t_{j+1} - t_j} \tag{D.25}$$

has been introduced to allow assumption of either *zero order hold* ($\boldsymbol{\alpha} = \boldsymbol{0}$) or *first order hold* ($\boldsymbol{\alpha} \neq \boldsymbol{0}$) on the inputs between sampling instants. The matrix exponential $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$ can be computed by means of a Padé approximation with repeated scaling and squaring (Moler and van Loan, 1978). However, both $\boldsymbol{\Phi}_s$ and the integral in (D.24) can be computed simultaneously through:

$$\exp\left(\begin{bmatrix} -\boldsymbol{A} & \boldsymbol{\sigma}\boldsymbol{\sigma}^T \\ \boldsymbol{0} & \boldsymbol{A}^T \end{bmatrix}\tau_s\right) = \begin{bmatrix} \boldsymbol{H}_1(\tau_s) & \boldsymbol{H}_2(\tau_s) \\ \boldsymbol{0} & \boldsymbol{H}_3(\tau_s) \end{bmatrix} \tag{D.26}$$

by combining submatrices of the result (van Loan, 1978), i.e.:

$$\boldsymbol{\Phi}_s = \boldsymbol{H}_3^T(\tau_s) \tag{D.27}$$

and:

$$\int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^T e^{\boldsymbol{A}^T s} ds = \boldsymbol{H}_3^T(\tau_s)\boldsymbol{H}_2(\tau_s) \tag{D.28}$$

Remark 3. The solution (D.23) to (D.20) is undefined if $\boldsymbol{A}$ is singular, but by introducing a coordinate transformation based on the SVD of $\boldsymbol{A}$ a solution to (D.20) can also be found for singular $\boldsymbol{A}$ (Kristensen *et al.*, 2002d).

D.2.2.2 Maximum a posteriori estimation

If prior information about the parameters is available in the form of a prior probability density function $p(\boldsymbol{\theta})$, Bayes' rule can be applied to give an improved estimate by forming the posterior probability density function:

$$p(\boldsymbol{\theta}|\mathcal{Y}_N) = \frac{p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{Y}_N)} \propto p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{D.29}$$

and subsequently finding the parameters that maximize this function, i.e. by performing *maximum a posteriori* (MAP) estimation. Assuming that the prior probability density of the parameters is Gaussian, and by introducing:

$$\boldsymbol{\mu}_{\boldsymbol{\theta}} = E\{\boldsymbol{\theta}\} \tag{D.30}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{\theta}} = V\{\boldsymbol{\theta}\} \tag{D.31}$$

and:

$$\boldsymbol{\epsilon}_{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}} \tag{D.32}$$

the posterior probability density function becomes:

$$\begin{aligned} p(\boldsymbol{\theta}|\mathcal{Y}_N) \propto & \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1}\boldsymbol{\epsilon}_k\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \\ & \times \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det\left(\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)}\left(\sqrt{2\pi}\right)^p} \end{aligned} \tag{D.33}$$

and the parameter estimates can now be determined by further conditioning on $\boldsymbol{y}_0$ and solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \left\{ -\ln\left(p(\boldsymbol{\theta}|\mathcal{Y}_N, \boldsymbol{y}_0)\right)\right\} \tag{D.34}$$

Remark 4. If no prior information is available (with $p(\boldsymbol{\theta})$ uniform), this formulation reduces to the ML formulation in (D.10), and MAP estimation can thus be seen as a generalization of ML estimation. In fact, the formulation also allows for MAP estimation on a subset of the parameters (with $p(\boldsymbol{\theta})$ partly uniform). Altogether, this increases the flexibility of the estimation scheme.

D.2.2.3 Using multiple independent data sets

If, instead of a single sequence of measurements, multiple consecutive, but yet separate, sequences of measurements, i.e. $\mathcal{Y}_{N_1}^1$, $\mathcal{Y}_{N_2}^2$, ..., $\mathcal{Y}_{N_i}^i$, ..., $\mathcal{Y}_{N_S}^S$, are available, a similar estimation method can be applied by expanding the expression for the posterior probability density function to the general form:

$$p(\boldsymbol{\theta}|\mathbf{Y}) \propto \left(\prod_{i=1}^{S} \left(\prod_{k=1}^{N_i} \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}^i\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0^i|\boldsymbol{\theta}) \right) \times \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det\left(\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)}\left(\sqrt{2\pi}\right)^p} \tag{D.35}$$

where:

$$\mathbf{Y} = [\mathcal{Y}_{N_1}^1, \mathcal{Y}_{N_2}^2, \ldots, \mathcal{Y}_{N_i}^i, \ldots, \mathcal{Y}_{N_S}^S] \tag{D.36}$$

and assuming the individual sequences to be stochastically independent. The parameter estimates can now be determined by further conditioning on:

$$\mathbf{y_0} = [\boldsymbol{y}_0^1, \boldsymbol{y}_0^2, \ldots, \boldsymbol{y}_0^i, \ldots, \boldsymbol{y}_0^S] \tag{D.37}$$

and applying nonlinear optimisation to find the minimum of the negative logarithm of the resulting posterior probability density function, i.e.:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \{-\ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right)\} \tag{D.38}$$

Remark 5. If only one sequence of measurements is available ($S = 1$), this formulation reduces to the MAP formulation in (D.34), and it can therefore be seen as a generalization of the MAP formulation for multiple independent data sets, which further increases the flexibility of the estimation scheme.

D.2.3 Data issues

Raw data sequences are often difficult to use for identification and parameter estimation purposes, e.g. if irregular sampling has been applied, if there are occasional outliers or if some of the observations are missing. The software implementation of the proposed estimation scheme (see Section D.3) also provides features to deal with these issues, and these features make it very flexible with respect to the types of data that can be used for the estimation.

D.2.3.1 Irregular sampling

The fact that the system equation (D.1) is formulated in continuous time makes it easy to deal with irregular sampling, because the corresponding state prediction equations of the EKF can be solved over time intervals of varying length.

D.2.3.2 Occasional outliers

The objective function (D.35) of the general formulation in (D.38) is quadratic in the innovations $\boldsymbol{\epsilon}_k^i$, and this means that the corresponding parameter estimates are heavily influenced by occasional outliers in the data sets used for the estimation. To deal with this problem a robust estimation method is applied, where the objective function is modified by replacing the quadratic term:

$$\nu_k^i = (\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i \tag{D.39}$$

with a threshold function $\varphi(\nu_k^i)$, which returns the argument for small values of ν_k^i, but is a linear function of $\boldsymbol{\epsilon}_k^i$ for large values of ν_k^i, i.e.:

$$\varphi(\nu_k^i) = \begin{cases} \nu_k^i & , \ \nu_k^i < c^2 \\ c(2\sqrt{\nu_k^i} - c) & , \ \nu_k^i \geq c^2 \end{cases} \tag{D.40}$$

where $c > 0$ is a constant. The derivative of this function with respect to $\boldsymbol{\epsilon}_k^i$ is a so-called influence function known as *Huber's ψ-function* (Huber, 1981).

D.2.3.3 Missing observations

The algorithms within the proposed estimation scheme make it easy to handle missing observations, i.e. to account for missing values in the output vector $\boldsymbol{y}_k^i$ when calculating, for some i and some k, the term:

$$\kappa_k^i = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}^i\right)}\left(\sqrt{2\pi}\right)^l} \tag{D.41}$$

in (D.35). The usual way to account for missing or non-informative values in the EKF is to set the corresponding elements of the covariance matrix $\boldsymbol{S}$ in (D.12) to infinity, which in turn gives zeroes in the corresponding elements of $(\boldsymbol{R}_{k|k-1})^{-1}$ and the Kalman gain matrix $\boldsymbol{K}_k$, meaning that no updating will take place in (D.15) and (D.16) corresponding to the missing values. This approach cannot be used for calculating (D.41), however, because a solution is needed which modifies $\boldsymbol{\epsilon}_k^i$ and $\boldsymbol{R}_{k|k-1}^i$ to reflect that the effective dimension of $\boldsymbol{y}_k^i$ is reduced due to the missing values. This is accomplished by replacing (D.2) with the alternative measurement equation:

$$\overline{\boldsymbol{y}}_k = \boldsymbol{E}\left(\boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k\right) \tag{D.42}$$

where $\boldsymbol{E}$ is an appropriate permutation matrix, which can be constructed from a unit matrix by eliminating the rows that correspond to the missing values in $\boldsymbol{y}_k$. If, for example, $\boldsymbol{y}_k$ has three elements, and the one in the middle is missing, the appropriate permutation matrix is given as follows:

$$\boldsymbol{E} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{D.43}$$

Equivalently, the regular equations of the EKF are replaced with the following alternative output prediction equations:

$$\hat{\overline{\boldsymbol{y}}}_{k|k-1} = \boldsymbol{E}\boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) \tag{D.44}$$

$$\overline{\boldsymbol{R}}_{k|k-1} = \boldsymbol{E}\boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{E}^T + \boldsymbol{E}\boldsymbol{S}\boldsymbol{E}^T \tag{D.45}$$

the alternative innovation equation:

$$\overline{\boldsymbol{\epsilon}}_k = \overline{\boldsymbol{y}}_k - \hat{\overline{\boldsymbol{y}}}_{k|k-1} \tag{D.46}$$

the alternative Kalman gain equation:

$$\overline{\boldsymbol{K}}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^T\boldsymbol{E}^T\overline{\boldsymbol{R}}_{k|k-1}^{-1} \tag{D.47}$$

and the alternative updating equations:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \overline{\boldsymbol{K}}_k\overline{\boldsymbol{\epsilon}}_k \tag{D.48}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \overline{\boldsymbol{K}}_k\overline{\boldsymbol{R}}_{k|k-1}\overline{\boldsymbol{K}}_k^T \tag{D.49}$$

The state prediction equations remain the same, and, with $\overline{l}$ being l minus the number of missing values in $\boldsymbol{y}_k^i$, this provides the necessary modifications of (D.41) to yield the following alternative term in (D.35):

$$\kappa_k^i = \frac{\exp\left(-\frac{1}{2}(\overline{\boldsymbol{\epsilon}}_k^i)^T(\overline{\boldsymbol{R}}_{k|k-1}^i)^{-1}\overline{\boldsymbol{\epsilon}}_k^i\right)}{\sqrt{\det\left(\overline{\boldsymbol{R}}_{k|k-1}^i\right)}\left(\sqrt{2\pi}\right)^{\overline{l}}} \tag{D.50}$$

D.2.4 Optimisation issues

To solve the nonlinear optimisation problem (D.38) a quasi-Newton method based on the BFGS updating formula and a soft line search algorithm is applied within the software implementation of the proposed estimation scheme (see Section D.3). This method is similar to the one presented by Dennis and Schnabel (1983), except for the fact that the gradient of the objective function here is approximated by a set of finite difference derivatives. During the initial iterations of the optimisation algorithm, *forward differences* are used, but as the minimum of the objective function is approached the algorithm shifts to *central differences* in order to reduce the error of the approximation.

In order to ensure stability in the calculation of the objective function in (D.38), simple constraints on the parameters are introduced, i.e.:

$$\theta_j^{\min} < \theta_j < \theta_j^{\max} \ , \ j = 1, \ldots, p \tag{D.51}$$

These constraints are satisfied by solving the optimisation problem with respect to a transformation of the original parameters, i.e.:

$$\tilde{\theta}_j = \ln\left(\frac{\theta_j - \theta_j^{\min}}{\theta_j^{\max} - \theta_j}\right) \ , \ j = 1, \ldots, p \tag{D.52}$$

A problem arises with this type of transformation when θ_j is very close to one of the limits, because the finite difference derivative with respect to θ_j may be close to zero, but this problem is solved by adding an appropriate penalty function to (D.38) to give the following modified objective function:

$$\mathcal{F}(\boldsymbol{\theta}) = -\ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right) + P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max}) \tag{D.53}$$

which is used instead. The penalty function is given as follows:

$$P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max}) = \lambda \left(\sum_{j=1}^{p} \frac{|\theta_j^{\min}|}{\theta_j - \theta_j^{\min}} + \sum_{j=1}^{p} \frac{|\theta_j^{\max}|}{\theta_j^{\max} - \theta_j} \right) \tag{D.54}$$

for $|\theta_j^{\min}| > 0$ and $|\theta_j^{\max}| > 0$, $j = 1, \ldots, p$. For proper choices of the Lagrange multiplier λ and the limiting values $\theta_j^{\min}$ and $\theta_j^{\max}$ the penalty function has no influence on the estimation when θ_j is well within the limits but will force the finite difference derivative to increase when θ_j is close to one of the limits.

D.2.5 Uncertainty of parameter estimates

Essential outputs of any statistical parameter estimation scheme include an assessment of the uncertainty of the estimates and quantities facilitating subsequent statistical tests. Within the software implementation of the proposed estimation scheme (see Section D.3), an estimate of the uncertainty of the parameter estimates is obtained by using the fact that by the central limit theorem the estimator in (D.38) is asymptotically Gaussian with mean $\boldsymbol{\theta}$ and covariance:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{H}^{-1} \tag{D.55}$$

where the matrix $\boldsymbol{H}$ is given by:

$$\{h_{ij}\} = -E\left\{ \frac{\partial^2}{\partial\theta_i \partial\theta_j} \ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right) \right\} \ , \ i, j = 1, \ldots, p \tag{D.56}$$

and where an approximation to $\boldsymbol{H}$ can be obtained from:

$$\{h_{ij}\} \approx -\left(\frac{\partial^2}{\partial\theta_i \partial\theta_j} \ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right) \right)\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \ , \ i, j = 1, \ldots, p \tag{D.57}$$

which is the Hessian evaluated at the minimum of the objective function. To obtain a measure of the uncertainty of the individual parameter estimates, the covariance matrix is decomposed as follows:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}} \boldsymbol{R} \boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}} \tag{D.58}$$

into $\boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}}$, which is a diagonal matrix of the standard deviations of the parameter estimates, and $\boldsymbol{R}$, which is the corresponding correlation matrix.

D.2.6 Statistical tests

The asymptotic Gaussianity of the estimator in (D.38) also allows marginal t-tests to be performed to test the hypothesis:

$$H_0: \quad \theta_j = 0 \tag{D.59}$$

against the corresponding alternative:

$$H_1: \quad \theta_j \neq 0 \tag{D.60}$$

i.e. to test whether a given parameter θ_j is marginally insignificant or not. The test quantity is the value of the parameter estimate divided by the standard deviation of the estimate, and under H_0 this quantity is asymptotically t-distributed with a number of degrees of freedom that equals the total number of observations minus the number of estimated parameters, i.e.:

$$z^t(\hat{\theta}_j) = \frac{\hat{\theta}_j}{\sigma_{\hat{\theta}_j}} \quad \in \quad t\left(\left(\sum_{i=1}^{S}\sum_{k=1}^{N_i} l\right) - p\right) \tag{D.61}$$

where, if there are missing observations in $\boldsymbol{y}_k^i$ for some i and some k, l is replaced with the appropriate value of $\overline{l}$. To facilitate these tests, $z^t(\hat{\theta}_j)$, $j = 1, \ldots, p$, are computed along with the following probabilities:

$$P\left(t<-|z^t(\hat{\theta}_j)| \wedge t>|z^t(\hat{\theta}_j)|\right) \ , j = 1, \ldots, p \tag{D.62}$$

D.3 Software implementation

The parameter estimation scheme presented in Section D.2 has been implemented in a software tool called **CTSM**, which is available for both Linux, Solaris and Windows platforms (Kristensen *et al.*, 2002d).

D.3.1 Features

Within the graphical user interface of **CTSM**, unknown parameters of model structures of the general type in (D.1)-(D.2) can be estimated using the methods presented in Section D.2. Once a model structure has been set up within the graphical user interface, the program analyzes the model equations to determine the symbolic names of the parameters and displays them to allow the user to specify which parameters to fix, which to estimate, and how each parameter should be estimated (ML or MAP). The program automatically generates and compiles the FORTRAN-code needed to perform the estimation, including the code for obtaining the Jacobians needed for linearization of the nonlinear equations (through analytical manipulation of the FORTRAN-code

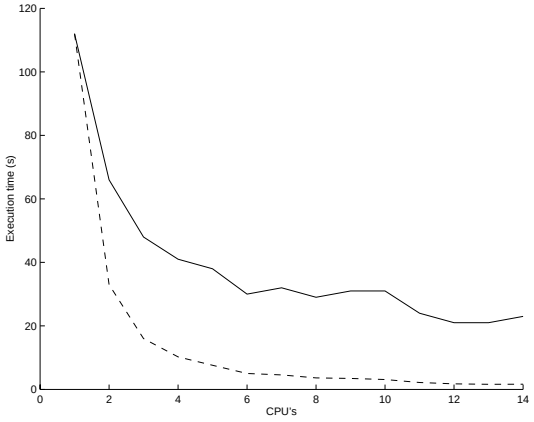


Figure D.1. Performance of **CTSM** when using shared memory parallelization. Solid lines: **CTSM** values; dashed lines: Theoretical values (linear scalability).

in a pre-compiler to avoid numerical approximation). After specifying which data sets to use, the program determines the parameter estimates and displays them along with the statistics mentioned in Section D.2. The program is very flexible with respect to the data sets that can be used for the estimation, because the features presented in Section D.2 for dealing with irregular sampling, occasional outliers and missing observations have all been implemented as well.

D.3.2 Shared memory parallelization

Estimating parameters in grey-box models is a computationally demanding task in general, and the estimation scheme presented in Section D.2 is no exception in this regard. On Solaris systems **CTSM** therefore supports shared memory parallelization using the OpenMP application program interface (API). More specifically, the finite difference derivatives of the objective function, which constitute the gradient approximation, can be computed in parallel.

Figure D.1 shows the performance benefits of this approach in terms of reduced execution time and demonstrates the scalability of the program for a small problem with 11 unknown parameters. The apparently non-existing effect of adding CPU's in the interval 6-10 is due to an uneven distribution of the workload (at least one CPU performs two finite difference computations, while the others wait), while for 11 and more CPU's the distribution is optimal.

D.4 Comparison with another software tool

A parameter estimation scheme rather similar to the one presented here and an associated software tool has previously been presented by Bohlin and Graebe (1995). There are, however, a number of very important differences between the two schemes, and this section is therefore devoted to outlining these differences and demonstrating their influence on the estimation performance of the corresponding software tools through comparative simulation studies.

As mentioned in Section D.3 the estimation scheme presented here has been implemented in a stand-alone tool called **CTSM**. The original tool incorporating the scheme of Bohlin and Graebe (1995) was called **IdKit**, but has been further developed into a more extensive tool called **MoCaVa** (Bohlin, 2001), which runs under MATLAB. Apart from parameter estimation, **MoCaVa** facilitates other important tasks within grey-box model development, e.g. model validation, and is superior to **CTSM** in that respect. The latter only allows state and output predictions to be computed based on a given data set, whereas the former has various test and visualization features that allow a given model to be tested on another data set or against other models using the same data set. In fact, the essence of **MoCaVa** is the ability to iteratively develop unfalsified models by means of such techniques, or, more specifically, by means of a method based on the stepwise forward inclusion rule and a modified likelihood ratio statistic (Bohlin and Graebe, 1995; Bohlin, 2001). However, for the purpose of the following comparison with **CTSM**, only parameter estimation will be considered, because this constitutes a fundamental information generating task, upon which subsequent model development can often be based.

D.4.1 Mathematical and algorithmic differences

Although very similar in terms of parameter estimation algorithms, there are some distinct differences between **MoCaVa** and **CTSM**. Generally, **MoCaVa** has more restrictions and uses more crude approximations than **CTSM** in order to reduce the computational burden at the expense of accuracy. The differences between the two tools are outlined in much more detail in the following.

D.4.1.1 General model structure

With respect to the general model structure, **MoCaVa** is less flexible than **CTSM**, primarily with respect to the diffusion term and the measurement noise term. Within **IdKit** the following class of models was allowed:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt + \boldsymbol{\sigma}(t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{D.63}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{D.64}$$

where $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(t_k, \boldsymbol{\theta}))$, i.e. almost the same class of models as in **CTSM**, but within **MoCaVa** this class has been restricted to the following:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt \tag{D.65}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{D.66}$$

where $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{\theta}))$ and $\boldsymbol{S}$ is a diagonal matrix. In other words, no diffusion term is allowed and there are more restrictions on the parameterization of the measurement noise term, which substantially limits flexibility. However, by instead allowing some of the input variables to be modelled as disturbances and by providing a library of generic disturbance models some of the flexibility has been retained. Indeed, Bohlin (2001) argues that moderately significant diffusion may be approximated quite well by a low-pass filtered white noise disturbance with a bandwidth that is slightly below the Nyquist frequency.

D.4.1.2 Parameter estimation methods

With respect to parameter estimation methods, both programs provide a ML estimation setup, but **MoCaVa** neither provides a MAP estimation setup nor allows estimation on multiple data sets as is the case with **CTSM**. Furthermore, the specific implementations of the ML estimation setup differ, although both programs rely on the same assumption of Gaussianity of the innovations and use the EKF to compute them. This is due to some important differences in the implementations of the EKF. **MoCaVa** uses an approach very similar to the linearization-based approach in **CTSM**, but without subsampling and with a more crude first order Taylor approximation to the matrix exponential, and, because diffusion terms are not allowed in the general model structure in **MoCaVa**, it suffices to compute the exponential of a much simpler matrix than in **CTSM**. Altogether, these differences reduce the computational load, but at the expense of accuracy. Even more importantly, like the original **IdKit** program, **MoCaVa** obtains the Jacobians needed for linearization of the non-linear equations by making finite difference approximations around a reference trajectory obtained by applying the EKF without updating. Thus the original equations are not linearized at points corresponding to the current state estimates, but at points along a deterministic reference trajectory. This is a very important difference from **CTSM**, which renders **IdKit** and hence **MoCaVa** unsuitable for estimation of parameters in systems with significant diffusion (Bohlin and Graebe, 1995; Bohlin, 2001) as demonstrated below.

D.4.1.3 Data issues

In terms of flexibility with respect to the types of data that can be used for the estimation, the two programs are almost equivalent. The only important difference is that **MoCaVa** does not incorporate any outlier robustness features, but relies on the user to remove outliers prior to the estimation.

D.4.1.4 Optimisation issues

There are also some important differences between the two programs with respect to optimisation method. **CTSM** uses a quasi-Newton method based on the BFGS updating formula for the Hessian and a soft line search algorithm, whereas **MoCaVa** uses a modified Newton-Raphson method, where the Hessian is approximated by applying a specific statistical assumption (Bohlin, 2001). Both programs use finite differences to approximate the gradient of the objective function, but **MoCaVa** only uses forward differences, while **CTSM** shifts from forward to central differences as the minimum is approached.

D.4.1.5 Uncertainty of parameter estimates

As opposed to **CTSM**, where an assessment of the uncertainty of the parameter estimates is obtained in terms of standard deviations of the estimates and their correlation matrix, no such information is obtained directly in **MoCaVa**.

D.4.1.6 Statistical tests

CTSM features simple marginal t-tests for significance of the individual parameters, whereas **MoCaVa** provides no such information at all.

D.4.2 Comparative simulation studies

In the following some of the effects of the differences between **MoCaVa** and **CTSM** are demonstrated with estimation results from two simulation examples.

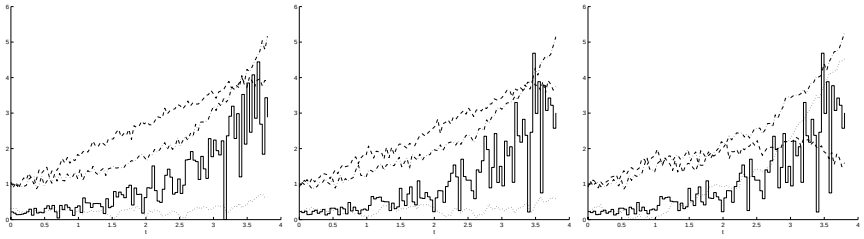
D.4.2.1 Example 1: Nonlinear (NL) model

The first example considered is a simple model of a fed-batch bioreactor. The system equation of this model is given in the following way:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \tag{D.67}$$

where X is the biomass concentration, S is the substrate concentration, V is the volume, F is the feed flow rate, $Y = 0.5$ is a yield coefficient and $S_F = 10$ is the feed concentration. The growth rate $\mu(S)$ is given as follows:

$$\mu(S) = \mu_{\max} \frac{S}{K_2 S^2 + S + K_1} \tag{D.68}$$



(a) No diffusion. (b) Weak diffusion. (c) Strong diffusion.

Figure D.2. Simulated data sets for the fed-batch bioreactor model in Example 1. Solid staircase: F; dashed lines: y_1; dotted lines: y_2; dash-dotted lines: y_3.

where $\mu_{\max}$, K_1 and $K_2 = 0.5$ are kinetic parameters. The corresponding measurement equation of the model is given in the following way:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k \ , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \ , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{D.69}$$

Using the true parameter and initial state values shown in Tables D.1-D.3 three different sets of data (101 samples each) were generated by stochastic simulation using the simple Euler scheme (Kloeden and Platen, 1992):

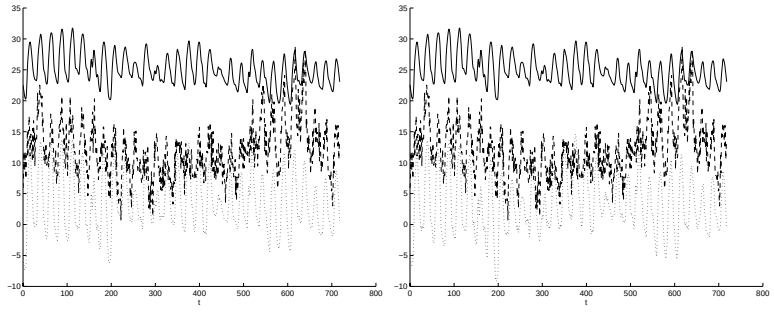
1. A data set with no diffusion (Figure D.2a).
2. A data set with weak diffusion (Figure D.2b).
3. A data set with strong diffusion (Figure D.2c).

Two sets of sparse versions of the same data sets were also generated by removing all y_2 measurements and subsequently all but every 10'th y_1 measurement.

D.4.2.2 Example 2: Linear time-invariant (LTI) model

The second example considered is a simple second order lumped parameter model of the heat dynamics of a wall with the following system equation:

$$\begin{aligned} d\begin{pmatrix} T_1 \\ T_2 \end{pmatrix} = (& \begin{bmatrix} -\frac{1}{G_1}\left(\frac{1}{H_1} + \frac{1}{H_2}\right) & \frac{1}{G_1 H_2} \\ \frac{1}{G_2 H_2} & -\frac{1}{G_2}\left(\frac{1}{H_2} + \frac{1}{H_3}\right) \end{bmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} \\ & + \begin{bmatrix} \frac{1}{G_1 H_1} & 0 \\ 0 & \frac{1}{G_2 H_3} \end{bmatrix} \begin{pmatrix} T_e \\ T_i \end{pmatrix}) dt + \begin{bmatrix} \sigma_{11} & 0 \\ 0 & \sigma_{22} \end{bmatrix} d\boldsymbol{\omega}_t \end{aligned} \tag{D.70}$$



(a) Without diffusion. (b) With diffusion.

Figure D.3. Simulated data sets for the lumped parameter wall heat dynamics model in Example 2. Solid lines: T_i; dashed lines: T_e; dotted lines: q_i.

where T_1 is the outer wall temperature, T_2 is the inner wall temperature, T_e is the outdoor temperature, T_i is the indoor temperature, and G_1, G_2, H_1, H_2 and H_3 are parameters of the second order thermal network describing the wall. The measurement equation of the model is given as follows:

$$(q_i)_k = \begin{bmatrix} 0 & -\frac{1}{H_3} \end{bmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}_k + \begin{bmatrix} 0 & \frac{1}{H_3} \end{bmatrix} \begin{pmatrix} T_e \\ T_i \end{pmatrix}_k + e_k \ , \quad e_k \in N(0, S) \tag{D.71}$$

Using the true parameter and initial state values shown in Tables D.4-D.5 two different sets of data (719 samples each) were again generated by stochastic simulation using the simple Euler scheme (Kloeden and Platen, 1992):

1. A data set without diffusion (Figure D.3a).
2. A data set with diffusion (Figure D.3b).

D.4.2.3 Quality of estimates

The first issue addressed in the comparison of the estimation performance of **MoCaVa** and **CTSM** is quality of estimates. A comparison of different estimators with respect to quality should ideally include an assessment of both bias and variance. However, since **MoCaVa** does not directly produce any information about the uncertainty of the parameter estimates, the two programs can only be compared in terms of bias. Tables D.1-D.3 show estimation results from both programs for the NL case in Example 1 using the data sets shown in Figure D.2. For the estimation in **MoCaVa** the diffusion term was approximated by a lowpass filtered white noise disturbance with a bandwidth of

Parameter	True value	**CTSM**	**MoCaVa**
X_0	1.0000E+00	1.0081E+00	9.9187E-01
S_0	2.4495E-01	2.5160E-01	2.3371E-01
V_0	1.0000E+00	1.0007E+00	9.9533E-01
$\mu_{\max}$	1.0000E+00	1.0104E+00	1.0143E+00
K_1	3.0000E-02	3.4177E-02	3.7176E-02
σ_{11}	0.0000E+00	6.8942E-06	9.9095E-03
σ_{22}	0.0000E+00	4.2411E-07	9.9727E-03
σ_{33}	0.0000E+00	5.1325E-07	9.7394E-03
S_{11}	1.0000E-02	9.0855E-03	8.6565E-03
S_{22}	1.0000E-03	9.7370E-04	9.4740E-04
S_{33}	1.0000E-02	9.4517E-03	8.9991E-03

Table D.1. Estimation results. Example 1 - Data in Figure D.2a.

Parameter	True value	**CTSM**	**MoCaVa**
X_0	1.0000E+00	9.8615E-01	9.9193E-01
S_0	2.4495E-01	2.3800E-01	2.3159E-01
V_0	1.0000E+00	9.7733E-01	1.0694E+00
$\mu_{\max}$	1.0000E+00	9.9694E-01	9.5656E-01
K_1	3.0000E-02	3.1506E-02	2.7128E-02
σ_{11}	1.0000E-01	1.1782E-01	3.0813E-01
σ_{22}	1.0000E-01	7.8251E-02	1.0167E-02
σ_{33}	1.0000E-01	6.2429E-02	1.0025E-02
S_{11}	1.0000E-02	8.0729E-03	9.2114E-03
S_{22}	1.0000E-03	9.2753E-04	1.2410E-03
S_{33}	1.0000E-02	9.3570E-03	1.2237E-02

Table D.2. Estimation results. Example 1 - Data in Figure D.2b.

Parameter	True value	**CTSM**	**MoCaVa**
X_0	1.0000E+00	9.6106E-01	9.5386E-01
S_0	2.4495E-01	2.3457E-01	1.0003E-01
V_0	1.0000E+00	9.9349E-01	1.0368E+00
$\mu_{\max}$	1.0000E+00	9.7142E-01	9.0460E-01
K_1	3.0000E-02	3.2600E-02	1.9886E-02
σ_{11}	3.1623E-01	3.2500E-01	1.1169E+00
σ_{22}	3.1623E-01	2.8063E-01	1.0046E-02
σ_{33}	3.1623E-01	2.6078E-01	5.5165E-01
S_{11}	1.0000E-02	7.7174E-03	9.9452E-03
S_{22}	1.0000E-03	1.1618E-03	1.1330E-02
S_{33}	1.0000E-02	8.3037E-03	1.5597E-02

Table D.3. Estimation results. Example 1 - Data in Figure D.2c.

10 rad/h (the Nyquist frequency is about 13.2 rad/h). The estimation results show that the estimates obtained with **CTSM** are less biased, in particular the estimates of the parameters of the diffusion term, some of which are an order of magnitude off in **MoCaVa**. Furthermore, the inability of **MoCaVa** to correctly estimate these parameters seems to introduce additional bias in the estimates of the other parameters for data sets with significant diffusion. Similar results have been obtained for the two sets of sparse versions of the same data sets. Tables D.4-D.5 show estimation results for the LTI case in Example 2 using the data sets shown in Figure D.3. For the estimation in **MoCaVa** the diffusion term was approximated by a lowpass filtered white noise disturbance with a bandwidth of 0.4 rad/h (the Nyquist frequency is 0.5 rad/h). In this case more similar estimates are obtained, except for the estimates of the parameters of the diffusion term, where **MoCaVa** again gives more bias.

Parameter	True value	**CTSM**	**MoCaVa**
T_{10}	1.3200E+01	1.3134E+01	1.3271E+01
T_{20}	2.5300E+01	2.5330E+01	2.5571E+01
G_1	1.0000E+02	1.0394E+02	1.0189E+02
G_2	5.0000E+01	4.9320E+01	4.9266E+01
H_1	1.0000E+00	9.6509E-01	9.8904E-01
H_2	2.0000E+00	2.0215E+00	1.9965E+00
H_3	5.0000E-01	5.0929E-01	5.0929E-01
σ_{11}	0.0000E+00	4.2597E-08	8.3838E-03
σ_{22}	0.0000E+00	1.4278E-09	5.1542E-03
S	1.0000E-02	1.0330E-02	1.0019E-02

Table D.4. Estimation results. Example 2 - Data in Figure D.3a.

Parameter	True value	**CTSM**	**MoCaVa**
T_{10}	1.3200E+01	1.9541E+01	1.4851E+01
T_{20}	2.5300E+01	2.5360E+01	2.5580E+01
G_1	1.0000E+02	1.0718E+02	7.6394E+01
G_2	5.0000E+01	5.3125E+01	5.4272E+01
H_1	1.0000E+00	1.9902E+00	1.4285E+00
H_2	2.0000E+00	9.0621E-01	1.9034E+00
H_3	5.0000E-01	5.0844E-01	5.1010E-01
σ_{11}	1.0000E-01	1.7791E-01	1.0206E-02
σ_{22}	1.0000E-01	1.4951E-01	1.4089E-01
S	1.0000E-02	9.4965E-03	3.2529E-02

Table D.5. Estimation results. Example 2 - Data in Figure D.3b.

D.4.2.4 Reproducibility

The second issue addressed in the comparison of the estimation performance of the two programs is reproducibility in terms of the sensitivity of the results to variations in initial values for the optimisation. Tables D.6-D.7 show estimation results from **CTSM** and **MoCaVa** respectively for the NL case corresponding to Table D.1 using four different sets of initial values. The initial values used are the true values shown in Table D.1, except for the values of the parameters of the diffusion term, which have been varied, ($[1, 0.1, 0.01, 0.001]$). The estimation results show that **MoCaVa** is much more sensitive than **CTSM** towards variations in initial values, particularly with respect to the parameters of the diffusion term. Tables D.8-D.9 show equivalent estimation results for the LTI case corresponding to Table D.4. The initial values used in this case are the true values shown in Table D.4, except for the values of the parameters of the diffusion term, which have again been varied ($[1, 0.1, 0.01, 0.001]$). Note that

Parameter	Result 1	Result 2	Result 3	Result 4
X_0	1.0081E+00	1.0081E+00	1.0081E+00	1.0086E+00
S_0	2.5160E-01	2.5160E-01	2.5160E-01	2.5205E-01
V_0	1.0007E+00	1.0007E+00	1.0007E+00	1.0006E+00
$\mu_{\max}$	1.0104E+00	1.0104E+00	1.0104E+00	1.0107E+00
K_1	3.4178E-02	3.4177E-02	3.4177E-02	3.4289E-02
σ_{11}	2.7167E-08	6.5411E-06	6.8942E-06	3.0674E-04
σ_{22}	3.5673E-06	8.7657E-18	4.2411E-07	5.9732E-05
σ_{33}	1.1250E-07	5.0250E-09	5.1325E-07	1.6944E-04
S_{11}	9.0855E-03	9.0855E-03	9.0855E-03	9.0844E-03
S_{22}	9.7371E-04	9.7370E-04	9.7370E-04	9.7068E-04
S_{33}	9.4517E-03	9.4517E-03	9.4517E-03	9.4239E-03

Table D.6. CTSM reproducibility. Example 1 - Data in Figure D.2a.

Parameter	Result 1	Result 2	Result 3	Result 4
X_0	9.8736E-01	9.8528E-01	9.9187E-01	9.9247E-01
S_0	2.5036E-01	2.3963E-01	2.3371E-01	2.3351E-01
V_0	1.0027E+00	9.9632E-01	9.9533E-01	9.9527E-01
$\mu_{\max}$	1.0230E+00	1.0213E+00	1.0143E+00	1.0134E+00
K_1	3.7723E-02	3.7639E-02	3.7176E-02	3.7035E-02
σ_{11}	1.4692E-01	6.2238E-02	9.9095E-03	9.9963E-04
σ_{22}	1.5229E-01	7.7283E-02	9.9727E-03	1.0000E-03
σ_{33}	1.2476E-01	5.8497E-02	9.7394E-03	1.0022E-03
S_{11}	8.2961E-03	8.4638E-03	8.6565E-03	8.6720E-03
S_{22}	9.0169E-04	9.3558E-04	9.4740E-04	9.4002E-04
S_{33}	8.7933E-03	8.8285E-03	8.9991E-03	9.0133E-03

Table D.7. MoCaVa reproducibility. Example 1 - Data in Figure D.2a.

for the first set of initial values, **MoCaVa** was not able to converge. Again the estimation results show that **MoCaVa** is more sensitive than **CTSM**, and again particularly with respect to the parameters of the diffusion term.

D.5 Discussion

The results presented in Section D.4 show that the software tool presented in Section D.3 for estimation of parameters in grey-box models (**CTSM**) generally performs well. In particular it performs significantly better than the one presented by Bohlin (2001) (**MoCaVa**) due to a number of algorithmic differences between the two programs, which have been pointed out.

In terms of quality of estimates, **CTSM** gives less bias than **MoCaVa**, especially with respect to the parameters of the diffusion term. It may be argued that this is due to the approximation used in **MoCaVa**, because the diffusion term cannot be modelled explicitly, and hence that a comparison should have

Parameter	Result 1	Result 2	Result 3	Result 4
T_{10}	1.3134E+01	1.3134E+01	1.3134E+01	1.3134E+01
T_{20}	2.5330E+01	2.5330E+01	2.5330E+01	2.5330E+01
G_1	1.0394E+02	1.0394E+02	1.0394E+02	1.0395E+02
G_2	4.9320E+01	4.9320E+01	4.9320E+01	4.9320E+01
H_1	9.6509E-01	9.6509E-01	9.6509E-01	9.6506E-01
H_2	2.0215E+00	2.0215E+00	2.0215E+00	2.0215E+00
H_3	5.0929E-01	5.0929E-01	5.0929E-01	5.0929E-01
σ_{11}	2.1538E-19	8.7694E-11	4.2597E-08	8.8565E-06
σ_{22}	3.4939E-08	5.5784E-08	1.4278E-09	3.0702E-07
S	1.0330E-02	1.0330E-02	1.0330E-02	1.0330E-02

Table D.8. CTSM reproducibility. Example 2 - Data in Figure D.3a.

Parameter	Result 1	Result 2	Result 3	Result 4
T_{10}	-	1.3070E+01	1.3271E+01	1.3168E+01
T_{20}	-	2.5577E+01	2.5571E+01	2.5567E+01
G_1	-	1.0270E+02	1.0189E+02	1.0373E+02
G_2	-	4.9277E+01	4.9266E+01	4.9312E+01
H_1	-	9.5979E-01	9.8904E-01	9.6833E-01
H_2	-	2.0277E+00	1.9965E+00	2.0180E+00
H_3	-	5.0935E-01	5.0929E-01	5.0929E-01
σ_{11}	-	2.2435E-02	8.3838E-03	9.9907E-04
σ_{22}	-	7.9109E-03	5.1542E-03	1.0036E-03
S	-	9.9315E-03	1.0019E-02	1.0224E-02

Table D.9. MoCaVa reproducibility. Example 2 - Data in Figure D.3a.

been made with the original **IdKit** program by Bohlin and Graebe (1995), but this program is not readily available. Furthermore, Bohlin and Graebe (1995) argue that **IdKit** cannot be expected to work properly for models with significant diffusion, so the differences in results from **CTSM** may be due to the construction of the algorithms after all. The specific algorithmic differences affecting the quality of the estimates are the more crude approximations made in **MoCaVa** in order to reduce the computational burden.

With respect to the quality of the estimates of the parameters of the diffusion term, it is particularly important that the EKF implementation in **CTSM** uses analytical Jacobians obtained at current values of the state estimates, whereas **MoCaVa** uses numerical Jacobians obtained at state values along a deterministic reference trajectory. This becomes particularly evident when comparing the results from the nonlinear model with the results from the linear time-invariant model. In the nonlinear case, **CTSM** performs significantly better than **MoCaVa**, whereas the two programs perform almost equally well in the linear time-invariant case, where the Jacobians are equal.

In terms of reproducibility, **CTSM** is less sensitive to initial values and hence gives more consistent results, which is most likely due to the gradient and Hessian approximations being more crude in the optimisation algorithm within **MoCaVa**. Evidence to support this conclusion is the fact that similar results have been obtained using data from a nonlinear as well as a linear time-invariant system without diffusion, indicating that the result is independent of the system type and of the diffusion term approximation mentioned above.

In the general context of providing support for systematic grey-box model development, **MoCaVa** is superior to **CTSM**, because of the additional features included to facilitate various model development tasks. In this context it may also be argued that the improvement in speed obtained through the approximations made in **MoCaVa** is an advantage, but unfortunately this improvement comes at the price of accuracy and consistency, particularly for the estimates of the parameters of the diffusion term. For applications where these are used directly, e.g. to assess the quality of a model (Kristensen *et al.*, 2001), to discriminate between models (Kristensen *et al.*, 2002a) or to pinpoint model deficiencies (Kristensen *et al.*, 2002c), one cannot afford to pay this price.

D.6 Conclusion

An efficient and flexible scheme for parameter estimation in stochastic grey-box models has been presented. The estimation scheme is based on the extended Kalman filter and features maximum likelihood as well as maximum a posteriori estimation on multiple independent data sets, including irregularly sampled data sets and data sets with occasional outliers and missing observations.

A software tool implementing the estimation scheme has also been presented and a comparison with an existing tool has indicated that the new tool has superior estimation performance both in terms of quality of estimates and in terms of reproducibility. In particular, the new tool provides more accurate and consistent estimates of the parameters of the diffusion term.

E

Paper no. 2

The paper[1] included in this appendix gives a condensed outline of the material presented in Chapter 2 in a more general context than modelling of fed-batch processes for the purpose of state estimation and optimal control. To be more specific, generalized versions of the grey-box modelling cycle and the corresponding algorithm are presented for modelling a variety of systems for different purposes. For illustration, the paper contains an extended version of the fed-batch bioreactor modelling example given in Chapter 2, which demonstrates that the proposed grey-box modelling framework can also be successfully applied, when all state variables of a model cannot be measured directly.

[1] The paper has been submitted for publication in *Computers and Chemical Engineering*.

A Method for Systematic Improvement of Stochastic Grey-Box Models

Niels Rode Kristensen[a], Henrik Madsen[b], Sten Bay Jørgensen[a]

[a]Department of Chemical Engineering, Technical University of Denmark, Building 229, DK-2800 Lyngby, Denmark

[b]Informatics and Mathematical Modelling, Technical University of Denmark, Building 321, DK-2800 Lyngby, Denmark

Abstract

A systematic framework for improving the quality of continuous time models of dynamic systems based on experimental data is presented. The framework is based on an interplay between stochastic differential equation modelling, statistical tests and nonparametric modelling and provides features that allow model deficiencies to be pinpointed and the structural origin of these deficiencies to be uncovered. More specifically, the proposed framework can be used to obtain estimates of unknown functional relations, in turn allowing unknown or inappropriately modelled phenomena to be uncovered. In this manner the framework permits systematic iterative model improvement. The performance of the proposed framework is illustrated with an example involving a dynamic model of a fed-batch bioreactor, where it is shown how an inappropriately modelled biomass growth rate can be uncovered and a proper functional relation inferred. A key point illustrated with this example is that functional relations involving variables that cannot be measured directly can also be uncovered.

Keywords: Model improvement; stochastic differential equations; parameter estimation; statistical tests; nonparametric modelling; bioreactor modelling.

E.1 Introduction

Dynamic process models are used in many areas of chemical engineering and for many different purposes. Dynamic model development is therefore inherently purpose-driven in the sense that the required accuracy of a model, in terms of prediction capabilities, depends on its intended application. More specifically, models intended for open-loop applications such as process simulation and optimisation, where long-term prediction capabilities are important, must be more accurate than models intended for closed-loop applications such as standard feedback control, where only short-term prediction capabilities are needed. However, to be more accurate, a model must be more complex, which means that it will be more difficult and time-consuming to develop. Finding a suitable model for a given purpose thus involves a trade-off between required model accuracy and affordable model complexity (Raisch, 2000).

For open-loop applications, ordinary differential equation (ODE) models or *white-box* models developed from first engineering principles and prior physical insights are typically used. Models of this type are often very detailed, because they must be able to capture nonlinear effects in order to be valid over wide ranges of state space, and, as a consequence, developing such models may be difficult and time-consuming. Indeed, the corresponding model development procedure is by no means guaranteed to converge, and few tools for making inferences about the proper structure of such models are available.

For closed-loop applications, much simpler input-output models or *black-box* models developed from experimental data with methods for time series analysis (Box and Jenkins, 1976) and system identification (Ljung, 1987; Söderström and Stoica, 1989) can often be used. Models of this type only have to be valid for a small range of state space, typically close to a constant operating point, which means that nonlinear effects can be neglected, making model development much faster. Furthermore, well-developed tools for structural identification of such linear models are available and the corresponding model development procedure is guaranteed to converge provided that certain conditions of identifiability of parameters and persistency of excitation of inputs are fulfilled.

Model-based optimizing control of batch and fed-batch processes, e.g. by means of nonlinear model predictive control (MPC) (Allgöwer and Zheng, 2000), represents a borderline case between open-loop and closed-loop applications, where neither of the above modelling approaches is ideally suited. On one hand, a model is needed, which is sufficiently accurate to be used for long-term prediction over wide ranges of state space, but on the other hand, the affordable model complexity is low due to the extreme importance of time-to-market issues in the biochemical, pharmaceutical and specialty chemicals industries, where batch and fed-batch processes are most commonly used.

A methodology that provides an appealing trade-off between the white-box and black-box approaches is *grey-box* modelling (Madsen and Melgaard, 1991;

Melgaard and Madsen, 1993; Bohlin and Graebe, 1995; Bohlin, 2001), where the key idea is to find the simplest model for a given purpose, which is consistent with prior physical knowledge and not falsified by available experimental data. In the approach by Bohlin and Graebe (1995) and Bohlin (2001) this is done by formulating a sequence of hypothetical model structures of increasing complexity and systematically expanding the model by falsifying incorrect hypotheses through statistical tests based on the experimental data. In this manner models can be developed, which have almost the same validity range as white-box models, but it can be done in a less time-consuming manner and the models being developed are guaranteed not to be overly complex.

Grey-box models are stochastic state space models consisting of a set of stochastic differential equations (SDE's) (Øksendal, 1998) describing the dynamics of the system in continuous time and a set of discrete time measurement equations. A considerable advantage of such models as opposed to white-box models is that they are designed to accommodate random effects. In particular, grey-box models allow for a decomposition of the noise affecting the system into a process noise term and a measurement noise term. As a consequence of this *prediction error decomposition* (PED), unknown parameters of grey-box models can be estimated from experimental data in a *prediction error* (PE) setting (Young, 1981), whereas for white-box models it can only be done in an *output error* (OE) setting (Young, 1981), which tends to give biased and less reproducible results, because random effects are absorbed into the parameter estimates, particularly if the model structure is inappropriate. Furthermore, PE estimation allows for a number of powerful statistical tools to be applied to provide indications for possible improvements to the model structure.

Grey-box modelling as presented by Bohlin and Graebe (1995) and Bohlin (2001) is an iterative and inherently interactive procedure, because it relies on the model maker to formulate the specific hypothetical model structures to be tested to improve the model. As pointed out by Bohlin (2001) this poses the problem that the model maker may run out of ideas for improvement before a sufficiently accurate model is obtained, which means that he or she may have to resort to using black-box models for filling the gaps in the model.

In the present paper a grey-box modelling framework is proposed, which relies less on the model maker. Within this framework specific model deficiencies can be pinpointed and their structural origin can be uncovered, which provides the model maker with valuable information about how to formulate new hypotheses to improve the model. This clearly speeds up the iterative model development procedure, and, as an additional benefit, also prevents the model maker from having to resort to using black-box models for filling the gaps in the models, when all prior physical knowledge is exhausted. The key to obtaining information about how to improve the model is the ability of the proposed framework to provide estimates of unknown functional relations, allowing unknown or inappropriately modelled phenomena to be uncovered. These estimates are obtained by making use of the PED and other properties of stochastic state space

models along with nonparametric modelling. The integration of nonparametric modelling with conventional grey-box modelling into a systematic framework for model improvement is the key contribution of this paper. The remainder of the paper is organized as follows: In Section E.2 the details of the proposed framework are outlined and in Section E.3 an example that illustrates its performance is presented. In Section E.4 a discussion of some important results is given and in Section E.5 the conclusions of the paper are presented.

E.2 Methodology

In this section the details of the proposed grey-box modelling framework are outlined. The overall framework is shown in Figure E.1 in the form of a modelling cycle, which shows the individual steps of the model development procedure. A key idea of grey-box modelling is to use all relevant prior physical knowledge, for which reason the first step within the modelling cycle is *model (re)formulation* based on first engineering principles, where the idea is to formulate an initial model structure (first modelling cycle iteration) or make modifications to this structure (subsequent iterations). The second step within the modelling cycle is *parameter estimation*, where the idea is to estimate unknown parameters of the model from available experimental data, and the third step is *residual analysis*, where the idea is to evaluate the quality of the resulting model by means of cross-validation. The fourth step within the modelling cycle is the important step of *model falsification or unfalsification*, which deals with whether or not, based on the available information, the model is sufficiently accurate to serve its intended purpose. If the model is unfalsified, the model development procedure can be terminated, but if the model is falsified, the modelling cycle must be repeated by re-formulating the model. A key feature of the proposed framework is that, in the latter case, the PED and other properties of stochastic state space models can be exploited to facilitate the task at hand. More specifically, the *statistical tests* of the fifth step within the modelling cycle can be applied to provide indications of which parts of the model that are deficient, and the *nonparametric modelling* techniques of the sixth step can be applied to provide estimates of the functional relations needed to repair these deficiencies to improve the model. In the remainder of this section the individual steps are described in more detail and an algorithm for systematic model improvement based on the proposed modelling cycle is presented.

E.2.1 Model (re)formulation

In the first step of the proposed grey-box modelling cycle, the idea is to formulate an initial model structure. This is a two-step procedure, because it involves derivation of a standard ODE model from first engineering principles and translation of the ODE model into a stochastic state space model consisting

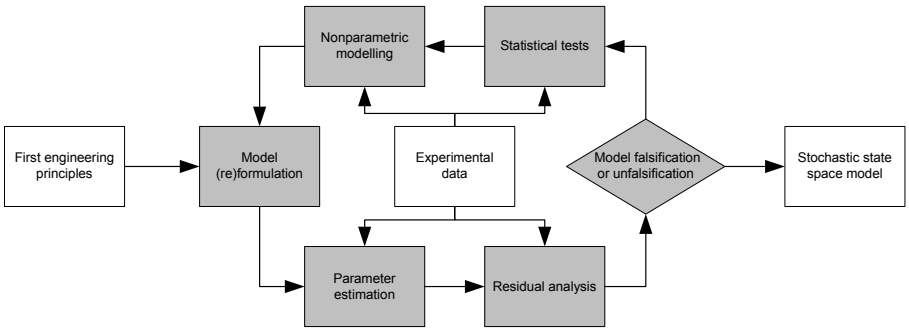


Figure E.1. The proposed grey-box modelling cycle. Boxes in grey illustrate tasks and boxes in white illustrate inputs to and outputs from the modelling cycle.

of a set of SDE's and a set of discrete time measurement equations. Deriving an ODE model from first engineering principles is a standard discipline for most chemical engineers and yields a model of the following type:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \tag{E.1}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathbb{R}^n$ is a vector of balanced quantities or state variables, $\boldsymbol{u}_t \in \mathbb{R}^m$ is a vector of input variables and $\boldsymbol{\theta} \in \mathbb{R}^p$ is a vector of possibly unknown parameters, and where $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$ is a nonlinear function. Translating the ODE model into a stochastic state space model is also relatively straightforward, because it can be done by replacing the ODE's with SDE's and adding a set of algebraic equations describing how measurements are obtained at discrete time instants. This yields a model of the following type:

$$d\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t \tag{E.2}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{E.3}$$

where $t \in \mathbb{R}$ is time, $\boldsymbol{x}_t \in \mathbb{R}^n$ is a vector of state variables, $\boldsymbol{u}_t \in \mathbb{R}^m$ is a vector of input variables, $\boldsymbol{y}_k \in \mathbb{R}^l$ is a vector of measured output variables, $\boldsymbol{\theta} \in \mathbb{R}^p$ is a vector of possibly unknown parameters, $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ and $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ are nonlinear functions, $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$.

The first term on the right-hand side of (E.2) is called the *drift* term and is a deterministic term equivalent to the term on the right-hand side of (E.1), whereas the second term on the right-hand side of (E.2) is called the *diffusion* term and is a stochastic term included to accommodate random effects due to e.g. approximation errors or unmodelled phenomena. A detailed account of the theory behind SDE's is given by Øksendal (1998). The diffusion term is the key to the proposed procedure for systematic model improvement, because

estimation of the parameters of this term from experimental data provides a measure of model uncertainty. The translation of the ODE model into a stochastic state space model does not affect the parameters of the drift term, which means that their physical interpretability is preserved.

Remark 1. The standard Wiener process $\{\boldsymbol{\omega}_t\}$, which drives the SDE's in (E.2), is a continuous stochastic process, which has stationary and independent time increments that are Gaussian and have zero mean and a covariance that is equal to the size of the time increment (Jazwinski, 1970).

Remark 2. The notation used in (E.2) is shorthand for the corresponding integral interpretation and is ambiguous unless a specific integral interpretation is given. SDE's may be interpreted in the sense of Stratonovich or in the sense of Itô (Jazwinski, 1970), but since the Stratonovich interpretation is unsuitable for parameter estimation (Åström, 1970), the Itô interpretation is adapted.

E.2.2 Parameter estimation

In the second step of the proposed modelling cycle the idea is to estimate the unknown parameters of the stochastic state space model (E.2)-(E.3) from experimental data. The solution to (E.2) is a Markov process, and an estimation scheme based on probabilistic methods can therefore be applied. A brief outline of the scheme used within the proposed framework is given in the following. A much more detailed account is given by Kristensen *et al.* (2002b).

E.2.2.1 Maximum likelihood (ML) estimation

Given a sequence of measurements $\boldsymbol{y}_0$, $\boldsymbol{y}_1$, ..., $\boldsymbol{y}_k$, ..., $\boldsymbol{y}_N$, ML estimates of the unknown parameters in (E.2)-(E.3) can be determined as the parameters $\boldsymbol{\theta}$ that maximize the likelihood function, i.e. the joint probability density:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = p(\mathcal{Y}_N|\boldsymbol{\theta}) = p(\boldsymbol{y}_N, \boldsymbol{y}_{N-1}, \ldots, \boldsymbol{y}_1, \boldsymbol{y}_0|\boldsymbol{\theta}) \tag{E.4}$$

or equivalently:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} p(\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}) \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{E.5}$$

where the rule $P(A \cap B) = P(A|B)P(B)$ has been applied to form a product of conditional probability densities. In order to obtain an exact evaluation of the likelihood function, a general nonlinear filtering problem must be solved (Jazwinski, 1970), but this is computationally infeasible in practice. However, since the increments of the standard Wiener process $\{\boldsymbol{\omega}_t\}$ driving the SDE's in (E.2) are Gaussian, it is reasonable to assume that the conditional probability densities in (E.5) can be well approximated by Gaussian densities. As a consequence, a method based on the much simpler extended Kalman filter (EKF)

can be applied (Kristensen *et al.*, 2002b). The Gaussian density is completely characterized by its mean and covariance, so by introducing the notation:

$$\hat{\boldsymbol{y}}_{k|k-1} = E\{\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}\} \tag{E.6}$$

$$\boldsymbol{R}_{k|k-1} = V\{\boldsymbol{y}_k|\mathcal{Y}_{k-1}, \boldsymbol{\theta}\} \tag{E.7}$$

and:

$$\boldsymbol{\epsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{E.8}$$

the likelihood function becomes:

$$L(\boldsymbol{\theta}; \mathcal{Y}_N) = \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1} \boldsymbol{\epsilon}_k\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{E.9}$$

and the parameter estimates can be determined by further conditioning on $\boldsymbol{y}_0$ and solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \{-\ln\left(L(\boldsymbol{\theta}; \mathcal{Y}_N|\boldsymbol{y}_0)\right)\} \tag{E.10}$$

where, for each set of parameters $\boldsymbol{\theta}$ in the optimisation, $\boldsymbol{\epsilon}_k$ and $\boldsymbol{R}_{k|k-1}$ are computed recursively by means of the EKF (Kristensen *et al.*, 2002b).

Remark 3. The validity of the Gaussianity assumption can (and should) be checked subsequent to the estimation (Holst *et al.*, 1992; Bak *et al.*, 1999).

E.2.2.2 Maximum a posteriori (MAP) estimation

If prior information about the parameters is available in the form of a prior probability density function $p(\boldsymbol{\theta})$, Bayes' rule can be applied to give an improved estimate by forming the posterior probability density function:

$$p(\boldsymbol{\theta}|\mathcal{Y}_N) = \frac{p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{Y}_N)} \propto p(\mathcal{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{E.11}$$

and subsequently finding the parameters that maximize this function, i.e. by performing MAP estimation. By assuming that the prior probability density of the parameters is Gaussian, and by introducing the notation:

$$\boldsymbol{\mu}_{\boldsymbol{\theta}} = E\{\boldsymbol{\theta}\} \tag{E.12}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{\theta}} = V\{\boldsymbol{\theta}\} \tag{E.13}$$

and:

$$\boldsymbol{\epsilon}_{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}} \tag{E.14}$$

the posterior probability density function becomes:

$$p(\boldsymbol{\theta}|\mathcal{Y}_N) \propto \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_k^T \boldsymbol{R}_{k|k-1}^{-1} \boldsymbol{\epsilon}_k\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \times \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det\left(\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)}\left(\sqrt{2\pi}\right)^p} \tag{E.15}$$

and the parameter estimates can now be determined by further conditioning on $\boldsymbol{y}_0$ and solving the following nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \left\{ -\ln\left(p(\boldsymbol{\theta}|\mathcal{Y}_N, \boldsymbol{y}_0)\right)\right\} \tag{E.16}$$

Remark 4. If no prior information is available ($p(\boldsymbol{\theta})$ uniform), this formulation reduces to the ML formulation in (E.10), and it can therefore be seen as a generalization of the ML formulation. In fact, this formulation also allows for MAP estimation on a subset of the parameters ($p(\boldsymbol{\theta})$ partly uniform).

E.2.2.3 Using multiple independent data sets

If multiple consecutive, but stochastically independent, sequences of measurements $\mathcal{Y}_{N_1}^1, \mathcal{Y}_{N_2}^2, \ldots, \mathcal{Y}_{N_i}^i, \ldots, \mathcal{Y}_{N_S}^S$, are available, a similar estimation method can be applied by expanding the posterior probability density function to:

$$p(\boldsymbol{\theta}|\mathbf{Y}) = p(\boldsymbol{\theta}|\mathcal{Y}_{N_1}^1, \mathcal{Y}_{N_2}^2, \ldots, \mathcal{Y}_{N_i}^i, \ldots, \mathcal{Y}_{N_S}^S]) \propto \left(\prod_{i=1}^{S} \left(\prod_{k=1}^{N_i} \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\epsilon}_k^i)^T (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\epsilon}_k^i\right)}{\sqrt{\det\left(\boldsymbol{R}_{k|k-1}^i\right)}\left(\sqrt{2\pi}\right)^l} \right) p(\boldsymbol{y}_0^i|\boldsymbol{\theta}) \right) \times \frac{\exp\left(-\frac{1}{2}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det\left(\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)}\left(\sqrt{2\pi}\right)^p} \tag{E.17}$$

and the parameter estimates can now be determined by further conditioning on $\mathbf{y_0} = [\boldsymbol{y}_0^1, \boldsymbol{y}_0^2, \ldots, \boldsymbol{y}_0^i, \ldots, \boldsymbol{y}_0^S]$ and solving the nonlinear optimisation problem:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \left\{ -\ln\left(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})\right)\right\} \tag{E.18}$$

Remark 5. If only one sequence of measurements is available ($S = 1$), this formulation reduces to the MAP formulation in (E.16), and it can therefore be seen as a generalization of this formulation for multiple independent data sets.

Kristensen *et al.* (2002b) give details about the estimation scheme used within the proposed framework, e.g. with respect to solving the nonlinear optimisation problem (E.18) and to robustness towards outliers and missing observations.

E.2.3 Residual analysis

In the third step of the proposed modelling cycle, the idea is to evaluate the quality of the model once the unknown parameters have been estimated.

An important aspect in assessing the quality of the model is to investigate its prediction capabilities by performing cross-validation and examining the corresponding residuals. Depending on the intended application of the model this should be done in either a one-step-ahead prediction setting (closed-loop applications) or in a pure simulation setting (open-loop applications). In either case a number of different methods can be applied (Holst *et al.*, 1992).

One of the most powerful of these methods is to compute and inspect the *sample autocorrelation function* (SACF) and the *sample partial autocorrelation function* (SPACF) (Brockwell and Davis, 1991) of the residuals to detect if they can be regarded as white noise or if there are significant lag dependencies, i.e. correlations between current and lagged values of the residuals, as this indicates that the prediction capabilities of the model are not perfect.

Nielsen and Madsen (2001a) recently presented extensions of these linear tools to nonlinear systems in the form of the *lag dependence function* (LDF) and the *partial lag dependence function* (PLDF), which are based on a close relation between correlation coefficients and the coefficients of determination for regression models. This relation allows for an extension to nonlinear systems by incorporating various nonparametric regression models.

Remark 6. Being an extension of the SACF, the LDF can be interpreted as being, for each lag k, the part of the overall variation in the observations of X_t from a stochastic process $\{X_t\}$, which can be explained by the observations of X_{t-k}. Likewise, being an extension of the SPACF, the PLDF can be interpreted as being, for each lag k, the relative decrease in one-step-ahead prediction variation when including X_{t-k} as an extra predictor.

Unlike the SACF and the SPACF, the LDF and the PLDF can also detect certain nonlinear lag dependencies and are therefore extremely useful for residual analysis within the proposed framework. More details about these and other similar tools are given by Nielsen and Madsen (2001a).

Remark 7. If the Gaussianity assumption mentioned in Section E.2.2 is valid, the statistical tests described in Section E.2.5 can also be applied in the evaluation of the quality of the model. However, the assumption is only likely to be valid, if the structure of the model is appropriate, which means that these tests should only be applied in the final stages of model development.

E.2.4 Model falsification or unfalsification

In the fourth step of the proposed modelling cycle, the idea is to determine whether or not, based on the information obtained in the previous step, the

model is sufficiently accurate to serve its intended purpose. This essentially involves a completely subjective decision by the model maker, addressing the trade-off between required model accuracy and affordable model complexity for the particular application. Nevertheless, a few guidelines can be given.

For models intended for closed-loop applications such as standard feedback control, where only short-term prediction capabilities are important, whiteness of cross-validation residuals obtained in a one-step-ahead prediction setting is a good indication of sufficient model accuracy. On the other hand, for models intended for open-loop applications such as process simulation and optimisation, where long-term prediction capabilities are important, whiteness of cross-validation residuals obtained in a pure simulation setting is a very good such indication. However, sufficient information may not be available to achieve this, which means that the model maker may have to settle for less.

If, with respect to the available information, the model is unfalsified for its intended purpose, the model development procedure can be terminated. If, on the other hand, the model is falsified, the modelling cycle must be repeated by re-formulating the model. In the latter case, the properties of the model in (E.2)-(E.3) facilitate the task at hand, however, as shown in the following.

E.2.5 Statistical tests

In the fifth step of the proposed modelling cycle, which is only needed if the model has been falsified and therefore needs to be improved, the idea is to apply statistical tests to provide indications of which parts of the model that are deficient. The statistical tests needed for this purpose are tests for significance of the individual parameters, particularly the parameters of the diffusion term.

Remark 8. If the residual sequences obtained in the third step of the modelling cycle can be regarded as stationary time series, the residual analysis tools mentioned in Section E.2.3 can also be applied in the analysis of possibilities for model improvement. More specifically, like the SACF and the SPACF, the LDF and the PLDF can be applied for structural identification (Nielsen and Madsen, 2001a), e.g. to determine if more state variables are needed.

An estimate of the uncertainty of the individual parameter estimates can be obtained by using the fact that by the central limit theorem the estimator in (E.18) is asymptotically Gaussian with mean $\boldsymbol{\theta}$ and covariance:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{H}^{-1} \tag{E.19}$$

where the matrix $\boldsymbol{H}$ is given by:

$$\{h_{ij}\} = -E\left\{\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln\left(p(\boldsymbol{\theta}|\mathbf{Y},\mathbf{y_0})\right)\right\}\ ,\ i,j=1,\ldots,p \tag{E.20}$$

and where an estimate of $\boldsymbol{H}$ can be obtained from:

$$\{h_{ij}\} \approx -\left(\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln\left(p(\boldsymbol{\theta}|\mathbf{Y},\mathbf{y_0})\right)\right)\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}},\ i,j=1,\ldots,p \tag{E.21}$$

which is the Hessian evaluated at the minimum of the objective function. To obtain a measure of the uncertainty of the individual parameter estimates, the covariance matrix can be decomposed as follows:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}}\boldsymbol{R}\boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}} \tag{E.22}$$

into $\boldsymbol{\sigma}_{\hat{\boldsymbol{\theta}}}$, which is a diagonal matrix of the standard deviations of the parameter estimates, and $\boldsymbol{R}$, which is the corresponding correlation matrix.

The asymptotic Gaussianity of the estimator in (E.18) also allows marginal t-tests to be performed to test the hypothesis:

$$H_0\text{:}\quad \theta_j = 0 \tag{E.23}$$

against the corresponding alternative:

$$H_1\text{:}\quad \theta_j \neq 0 \tag{E.24}$$

i.e. to test whether a given parameter θ_j is marginally insignificant or not. The test quantity is the value of the parameter estimate divided by the standard deviation of the estimate, and under H_0 this quantity is asymptotically t-distributed with a number of degrees of freedom that equals the total number of observations minus the number of estimated parameters, i.e.:

$$z^t(\hat{\theta}_j) = \frac{\hat{\theta}_j}{\sigma_{\hat{\theta}_j}} \quad \in \quad t\left(\left(\sum_{i=1}^{S}\sum_{k=1}^{N_i} l\right) - p\right) \tag{E.25}$$

Due to correlations between the individual parameter estimates, a series of such marginal tests cannot be used to test the hypothesis that a subset of the parameters, $\boldsymbol{\theta}_* \subset \boldsymbol{\theta}$, are simultaneously insignificant:

$$H_0\text{:}\quad \boldsymbol{\theta}_* = \mathbf{0} \tag{E.26}$$

against the alternative that they are not:

$$H_1\text{:}\quad \boldsymbol{\theta}_* \neq \mathbf{0} \tag{E.27}$$

Hence a test that takes correlations into account must be used instead, e.g. a likelihood ratio test, a Lagrange multiplier test or a test based on Wald's W-statistic (Holst *et al.*, 1992). Under H_0 the test quantities for these tests all have the same asymptotic χ^2-distribution with a number of degrees of freedom that equals the number of parameters subjected to the test (Holst *et al.*, 1992), but in the context of the proposed framework the test based on Wald's

W-statistic has the advantage that no re-estimation of the parameters is required, because it can simply be computed in the following way:

$$W(\hat{\boldsymbol{\theta}}_*) = \hat{\boldsymbol{\theta}}_*^T \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}_*}^{-1} \hat{\boldsymbol{\theta}}_* \quad \in \quad \chi^2\left(\dim(\hat{\boldsymbol{\theta}}_*)\right) \tag{E.28}$$

where $\hat{\boldsymbol{\theta}}_* \subset \hat{\boldsymbol{\theta}}$ is the subset of the parameter estimates subjected to the test and $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}_*}$ is the covariance matrix of these estimates. This covariance matrix can be computed from the full covariance matrix as follows:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}_*} = \boldsymbol{E}\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}\boldsymbol{E}^T \tag{E.29}$$

where $\boldsymbol{E}$ is a permutation matrix constructed from a unit matrix by eliminating the rows that correspond to parameter estimates not subjected to the test.

Remark 9. Strictly speaking, these tests should only be applied if the Gaussianity assumption mentioned in Section E.2.2 is valid, which is only likely to be the case in the final stages of model development, where the structure of the model is appropriate. Nevertheless, the corresponding test results can be used to provide reasonable indications for model improvement.

The above tests for insignificance provide the necessary framework for obtaining indications of which parts of the model that are deficient. In principle, *insignificant* parameters are parameters that may be eliminated, and, generally, the presence of such parameters is therefore an indication that the model is overparameterized. On the other hand, because of the particular nature of the model in (E.2)-(E.3), where the diffusion term is included to account for random effects due to e.g. approximation errors or unmodelled phenomena, the presence of *significant* parameters in the diffusion term is an indication that the corresponding drift term may be incorrect, which in turn provides an uncertainty measure that allows model deficiencies to be detected. If, instead of the general parameterization of the diffusion term indicated in (E.2), a diagonal parameterization is used, this also allows the deficiencies to be pinpointed in the sense that deficiencies in specific elements of the drift term can be detected.

E.2.5.1 Pinpointing model deficiencies

If a diagonal parameterization of the diffusion term in (E.2) is used, the presence of significant parameters in a given diagonal element is an indication that the corresponding element of the drift term may be incorrect. This is valuable information for the model maker, as it indicates that some of the inherent phenomena of this term may be inappropriately modelled. If, by using physical insights, the model maker is able to subsequently select a specific phenomena model for further analysis, the proposed framework also provides means to confirm the suspicion that this model is inappropriate, if it is in fact true.

Typical suspect phenomena models include models of reaction rates, heat and mass transfer rates and similar complex dynamic phenomena, all of which can usually be described using functions of the state and input variables, i.e.:

$$r_t = \varphi(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta}) \tag{E.30}$$

where r_t symbolizes the phenomenon of interest and $\varphi(\cdot) \in \mathbb{R}$ is the nonlinear function used by the model maker to describe it. To confirm the suspicion that $\varphi(\cdot)$ is inappropriate, the parameter estimation step must be repeated with a re-formulated version of the model in (E.2)-(E.3) to give new statistical information. More specifically, if r_t is isolated by including it in the re-formulated model as an additional state variable, i.e.:

$$d\boldsymbol{x}_t^* = \boldsymbol{f}^*(\boldsymbol{x}_t^*, \boldsymbol{u}_t, t, \boldsymbol{\theta})dt + \boldsymbol{\sigma}^*(\boldsymbol{u}_t, t, \boldsymbol{\theta})d\boldsymbol{\omega}_t^* \tag{E.31}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k^*, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k \tag{E.32}$$

where $\boldsymbol{x}_t^* = [\boldsymbol{x}_t^T \; r_t]^T$, $\boldsymbol{\sigma}^*(\cdot) \in \mathbb{R}^{(n+1)\times(n+1)}$ and $\{\boldsymbol{\omega}_t^*\}$ is an $(n+1)$-dimensional standard Wiener process and where:

$$\boldsymbol{f}^*(\boldsymbol{x}_t^*, \boldsymbol{u}_t, t, \boldsymbol{\theta}) = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta}) \\ \frac{\partial \varphi(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta})}{\partial \boldsymbol{x}_t} \frac{d\boldsymbol{x}_t}{dt} + \frac{\partial \varphi(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta})}{\partial \boldsymbol{u}_t} \frac{d\boldsymbol{u}_t}{dt} \end{pmatrix} \tag{E.33}$$

the presence of significant parameters in the corresponding diagonal element of the expanded diffusion term is a strong indication that $\varphi(\cdot)$ is inappropriate.

Remark 10. A particularly simple but nevertheless very important special case of the above formulation is obtained if $\varphi(\cdot)$ is assumed to be constant, in which case the partial derivatives in (E.33) are both zero and any variation in r_t must be explained by the corresponding diagonal element of the expanded diffusion term, which in turn means that if the parameters of this diagonal element are significant, this is an indication that $\varphi(\cdot)$ is not constant.

E.2.6 Nonparametric modelling

In the sixth step of the proposed modelling cycle, which can only be used if specific model deficiencies have been pinpointed as described above, the idea is to uncover the structural origin of these deficiencies. The procedure for accomplishing this is based on a combination of the applicability of stochastic state space models for state estimation and the ability of nonparametric regression methods to provide visualizable estimates of unknown functional relations.

E.2.6.1 Estimating unknown functional relations

Using the re-formulated model in (E.31)-(E.32) and the corresponding parameter estimates, state estimates $\hat{\boldsymbol{x}}_{k|k}^*$, $k = 0, \ldots, N$, can be obtained for a given

set of experimental data by applying the EKF. In particular, since the inappropriately modelled phenomenon r_t is included as an additional state variable in this model, estimates $\hat{r}_{k|k}$, $k = 0, \ldots, N$, can be obtained, which in turn facilitates application of nonparametric regression to provide estimates of possible functional relations between r_t and the state and input variables.

Several nonparametric regression techniques are available (Hastie *et al.*, 2001), but in the context of the proposed framework, *additive models* (Hastie and Tibshirani, 1990) are preferred, because fitting such models circumvents the *curse of dimensionality*, which tends to render nonparametric regression infeasible in higher dimensions, and because results obtained with such models are particularly easy to visualize, which is also important.

Remark 11. Additive models are nonparametric extensions of linear regression models and are fitted by using a training data set of observations of several predictor variables $X_1, \ldots, X_n$ and a single response variable Y to compute a smoothed estimate of the response variable for a given set of values of the predictor variables. This is done by assuming that the contributions from each of the predictor variables are additive and can be fitted nonparametrically using the *backfitting algorithm* (Hastie and Tibshirani, 1990).

Using additive models, the variation in r_t can be decomposed into the variation that can be attributed to each of the state and input variables in turn, and the result can be visualized by means of partial dependence plots with associated bootstrap confidence intervals (Hastie *et al.*, 2001). In this manner it may be possible to reveal the true structure of the function describing r_t, i.e.:

$$r_t = \varphi_{\text{true}}(\boldsymbol{x}_t, \boldsymbol{u}_t, \boldsymbol{\theta}) \tag{E.34}$$

which in turn provides the model maker with valuable information about how to re-formulate the model for the next modelling cycle iteration. Needless to say, this should be done in accordance with physical insights.

Remark 12. The assumption of additive contributions does not necessarily limit the ability of additive models to reveal non-additive functional relations involving more than one predictor variable, since, by proper processing of the training data set, functions of more than one predictor variable, e.g. $X_1 X_2$, can be included as predictor variables as well (Hastie and Tibshirani, 1990).

E.2.7 An algorithm for systematic model improvement

In the following the methodologies from the various steps of the proposed modelling cycle are summarized in the form of an algorithm for systematic model improvement given a pre-specified purpose of the model:

1. Use first engineering principles and physical insights to derive an initial model structure in the form of an ODE model (see Section E.2.1).

2. Translate the ODE model into a stochastic state space model using a diagonal parameterization of the diffusion term (see Section E.2.1).

3. Estimate the parameters of the model from available experimental data using ML or MAP estimation (see Section E.2.2).

4. Evaluate the quality of the resulting model by performing residual analysis on cross-validation data (see Section E.2.3).

5. Determine if the model is sufficiently accurate to serve its intended purpose. If unfalsified, terminate model development. If falsified, proceed with model development (see Section E.2.4).

6. Try to pinpoint specific model deficiencies by applying statistical tests and by re-formulating the model with additional state variables and repeating the estimation and test procedures (see Section E.2.5).

7. If specific model deficiencies can be pinpointed, use state estimation and nonparametric modelling to uncover their structural origin by obtaining appropriate estimates of functional relations (see Section E.2.6).

8. Re-formulate the model according to the estimated functional relations and physical insights and repeat from Step 3 (see Section E.2.6).

This algorithm can be applied to develop new as well as to improve existing models of dynamic systems for a variety of purposes. More specifically, models can be developed with emphasis on short-term as well as long-term prediction capabilities, i.e. models intended for closed-loop as well as open-loop applications. However, as further discussed in Section E.4, the algorithm is not guaranteed to converge, especially not if insufficient prior information is available or if the quality and amount of available experimental data is limited.

In particular, a situation may occur, where the model is falsified, but where none of the parameters of the diffusion term appear to be significant and pinpointing a specific model deficiency is impossible. A situation may also occur, where the model is falsified and the significance of certain parameters of the diffusion term have allowed a specific deficiency to be pinpointed, but where the structural origin of the deficiency cannot be uncovered. In the context of the proposed framework, both situations imply that a point has been reached, where the model cannot be further improved with the available information.

Remark 13. The estimation methods described in Section E.2.2 (estimation in a PE setting) tend to emphasize the one-step-ahead prediction capabilities of the model and are therefore not ideal for models intended for open-loop applications. Nevertheless, these methods should be used in the development of such models as well, because of the possibility of using the tools described above for improving the structure of the model, if necessary, which would otherwise not be possible. Once an appropriate model structure has been obtained

(ultimately corresponding to an insignificant diffusion term), the parameters should then be re-calibrated with an estimation method that emphasizes the pure simulation capabilities of the model (estimation in an OE setting).

E.3 Example: Modelling a fed-batch bioreactor

To illustrate the performance of the proposed framework in terms of improving the quality of an existing model, a simple simulation example is considered in the following. The process considered is a fed-batch bioreactor, where the true model used for simulation of the process is given in the following way:

$$\frac{dX}{dt} = \mu(S)X - \frac{FX}{V} \tag{E.35}$$

$$\frac{dS}{dt} = -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \tag{E.36}$$

$$\frac{dV}{dt} = F \tag{E.37}$$

where X is the biomass concentration, S is the substrate concentration, V is the volume, F is the feed flow rate, $Y = 0.5$ is the yield coefficient of biomass, $S_F = 10$ is the feed concentration of substrate, and $\mu(S)$ is the biomass growth rate, which is described by Monod kinetics with substrate inhibition, i.e.:

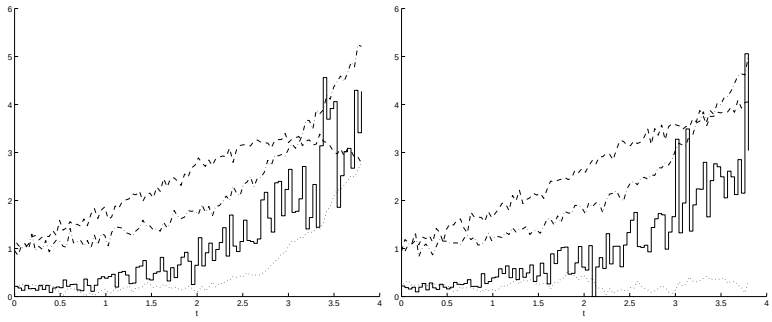
$$\mu(S) = \mu_{\max}\frac{S}{K_2S^2 + S + K_1} \tag{E.38}$$

where $\mu_{\max} = 1$, $K_1 = 0.03$ and $K_2 = 0.5$. Using $(X_0, S_0, V_0) = (1, 0.2449, 1)$ as initial states, simulation data sets from two batch runs (101 samples each) are generated by perturbing the feed flow rate along a pre-determined trajectory and subsequently adding Gaussian measurement noise to the appropriate variables using the noise levels mentioned beneath Figure E.2.

In the following it is assumed that the model to be developed is to be used for an open-loop application, where long-term prediction capabilities are important, and that the model maker has been able to set up an initial model structure corresponding to (E.35)-(E.37) but is unaware of the true structure of $\mu(S)$ given in (E.38). In terms of available measurements, two different cases are considered: A full state information case, where it is assumed that all state variables can be measured, and a partial state information case, where it is assumed that only the biomass and the volume can be measured.

E.3.1 Case 1: Full state information

The available sets of experimental data for the full state information case are shown in Figure E.2. Using these data sets it will now be illustrated how the



(a) Batch no. 1. (b) Batch no. 2.

Figure E.2. The two batch data sets available for case 1. Solid staircase: Feed flow rate F; dashed lines: Biomass measurements y_1 (with $N(0, 0.01)$ noise); dotted lines: Substrate measurements y_2 (with $N(0, 0.001)$ noise); dash-dotted lines: Volume measurements y_3 (with $N(0, 0.01)$ noise)).

proposed modelling cycle can be used to improve the initial model set up by the model maker. In this particular case only two iterations of the modelling cycle are needed. In the general case more iterations may be needed.

E.3.1.1 First modelling cycle iteration

Model formulation

The first iteration of the modelling cycle starts with the model formulation step, where it is assumed that the model maker has been able to set up an initial model structure corresponding to (E.35)-(E.37), which is then translated into a stochastic state space model with the following system equation:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -\frac{\mu X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt \; + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \tag{E.39}$$

and the following measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \; \boldsymbol{e}_k \; , \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \; , \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix} \tag{E.40}$$

where, because the true structure of $\mu(S)$ given in (E.38) is unknown, a constant biomass growth rate μ has been assumed. As recommended above, a diagonal

parameterization of the diffusion term in the system equation has been used to allow model deficiencies to be pinpointed if the model is falsified.

Parameter estimation

As the next step, the unknown parameters of the model in (E.39)-(E.40) are estimated by means of the ML method using the data from batch no. 1 (Figure E.2a), which gives the results shown in Table E.1.

Residual analysis

Evaluating the quality of the resulting model as the next step, cross-validation residual analysis is performed as shown in Figure E.3. This analysis shows that the model does a poor job in pure simulation, particularly for y_1 and y_2, whereas its one-step-ahead prediction capabilities are quite good.

Model falsification or unfalsification

Moving to the model falsification or unfalsification step, the poor pure simulation capabilities falsify the model for its intended purpose, which means that the modelling cycle must be repeated by re-formulating the model.

Statistical tests

To obtain information about how to re-formulate the model in an intelligent way, model deficiencies should be pinpointed. Table E.1 also includes t-scores for performing marginal tests for insignificance of the individual parameters, which show that, on a 5% level, only one of the parameters of the diffusion term is insignificant, i.e. σ_{33}, whereas σ_{11} and σ_{22} are both significant, which

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.6973E-01	3.4150E-02	28.3962	Yes
S_0	2.5155E-01	3.1938E-02	7.8761	Yes
V_0	1.0384E+00	1.8238E-02	56.9359	Yes
μ	6.8548E-01	2.2932E-02	29.8921	Yes
σ_{11}	1.8411E-01	2.5570E-02	7.2000	Yes
σ_{22}	2.2206E-01	3.4209E-02	6.4912	Yes
σ_{33}	2.7979E-02	1.7943E-02	1.5594	No
S_{11}	6.7468E-03	1.3888E-03	4.8580	Yes
S_{22}	3.9131E-04	2.4722E-04	1.5828	No
S_{33}	1.0884E-02	1.5409E-03	7.0633	Yes

Table E.1. Estimation results. Model in (E.39)-(E.40) - data from Figure E.2a.

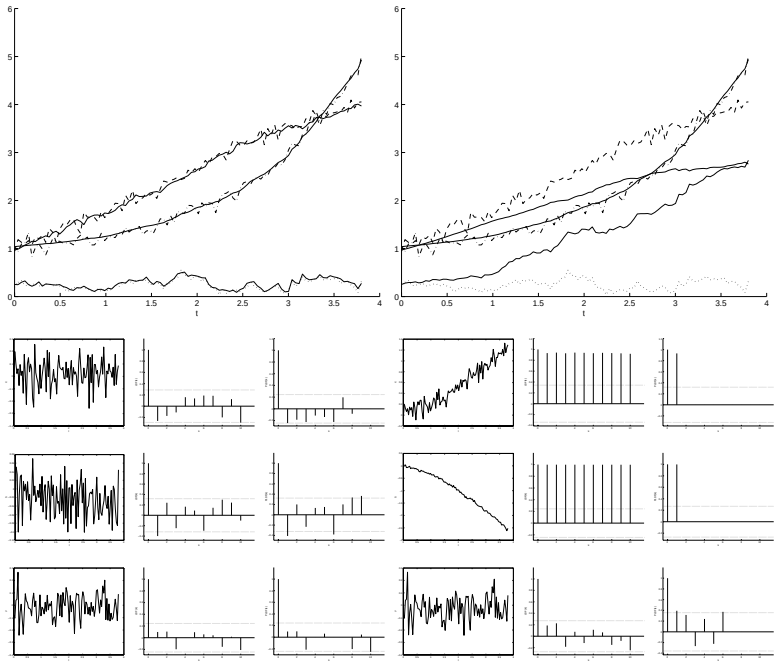
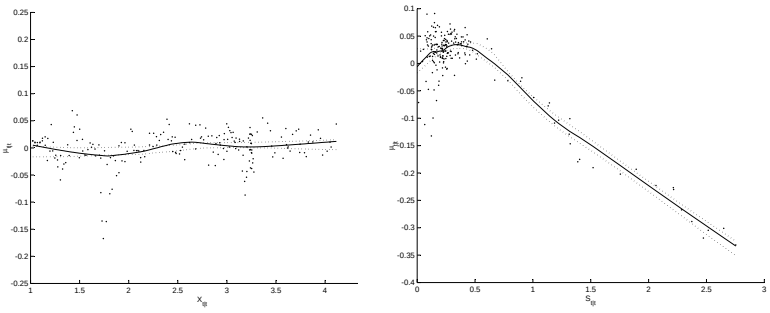


Figure E.3. Cross-validation residual analysis results for the model in (E.39)-(E.40) with parameters in Table E.1 using the data from batch no. 2 (Figure E.2b). Top left: One-step-ahead prediction comparison (solid lines: Predicted values); top right: Pure simulation comparison (solid lines: Simulated values); bottom left: One-step-ahead prediction residuals, LDF and PLDF for y_1, y_2 and y_3; bottom right: Pure simulation residuals, LDF and PLDF for y_1, y_2 and y_3.

indicates that the first two elements of the drift term may be incorrect. These elements both depend on μ and a skilled model maker, who knows how difficult it is to model complex dynamic phenomena such as biomass growth, would immediately suspect μ to be deficient. To avoid jumping to conclusions, the suspicion should be confirmed, which is done by first re-formulating the model with μ as an additional state variable, which yields the system equation:

$$d\begin{pmatrix} X \\ S \\ V \\ \mu \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -\frac{\mu X}{Y} + \frac{F(S_F - S)}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \qquad \text{(E.41)}$$

where, because μ has been assumed to be constant, the last element of the drift term is zero. The measurement equation is the same as in (E.40). Estimating



(a) $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$ (b) $\hat{\mu}_{k|k}$ vs. $\hat{S}_{k|k}$

Figure E.4. Partial dependence plots of $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$. Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals (1000 replicates).

the parameters of this model, using the same data set as before, gives the results shown in Table E.2, and inspection of the t-scores for marginal tests for insignificance now show that, of the parameters of the diffusion term, only σ_{44} is significant on a 5% level. This in turn indicates that there is substantial variation in μ and thus confirms the suspicion that μ is deficient.

Nonparametric modelling

Having pinpointed μ as being deficient, nonparametric modelling can be applied as the next step to uncover the structural origin of the deficiency. Using

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0239E+00	4.9566E-03	206.5723	Yes
S_0	2.3282E-01	1.1735E-02	19.8405	Yes
V_0	1.0099E+00	3.8148E-03	264.7290	Yes
μ_0	7.8658E-01	2.4653E-02	31.9061	Yes
σ_{11}	2.0791E-18	1.4367E-17	0.1447	No
σ_{22}	1.1811E-30	1.6162E-29	0.0731	No
σ_{33}	3.1429E-04	2.0546E-04	1.5297	No
σ_{44}	1.2276E-01	2.5751E-02	4.7674	Yes
S_{11}	7.5085E-03	9.9625E-04	7.5368	Yes
S_{22}	1.1743E-03	1.6803E-04	6.9887	Yes
S_{33}	1.1317E-02	1.3637E-03	8.2990	Yes

Table E.2. Estimation results. Model in (E.41) and (E.40) - data from Figure E.2a.

the re-formulated model in (E.41) and (E.40) and the parameter estimates in Table E.2, state estimates $\hat{X}_{k|k}$, $\hat{S}_{k|k}$, $\hat{V}_{k|k}$, $\hat{\mu}_{k|k}$, $k = 0, \ldots, N$, are obtained by means of the EKF and an additive model is fitted to reveal the true structure of the function describing μ by means of estimates of functional relations between μ and the state and input variables. It is reasonable to assume that μ does not depend on V and F, so only functional relations between $\hat{\mu}_{k|k}$ and $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$ are estimated, which gives the results shown in Figure E.4 in the form of partial dependence plots with associated bootstrap confidence intervals. These plots indicate that $\hat{\mu}_{k|k}$ does not depend on $\hat{X}_{k|k}$, but is highly dependent on $\hat{S}_{k|k}$, which in turn suggests to replace the assumption of constant μ with an assumption of μ being a function of S, when the model is re-formulated for the next iteration of the modelling cycle. More specifically, this function should somehow comply with the functional relation revealed in Figure E.4b.

E.3.1.2 Second modelling cycle iteration

Model re-formulation

To a skilled model maker with experience in bioreactor modelling, the functional relation revealed in the partial dependence plot between $\hat{\mu}_{k|k}$ and $\hat{S}_{k|k}$ in Figure E.4 is a clear indication that the growth of biomass is governed by Monod kinetics and inhibited by substrate, which in the first step of the second iteration of the modelling cycle makes it possible to re-formulate the model in (E.39)-(E.40) accordingly to yield the following system equation:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \tag{E.42}$$

where $\mu(S)$ is given by the true structure in (E.38). The measurement equation of course remains unchanged and is therefore the same as in (E.40).

Parameter estimation

As the next step, estimation of the unknown parameters of the re-formulated model using the same data set as before gives the results shown in Table E.3.

Residual analysis

Evaluating the quality of the resulting model is the next step. Cross-validation residual analysis is therefore performed as shown in Figure E.5, and the results of this analysis show that the one-step-ahead prediction capabilities as well as the pure simulation capabilities of the re-formulated model are very good.

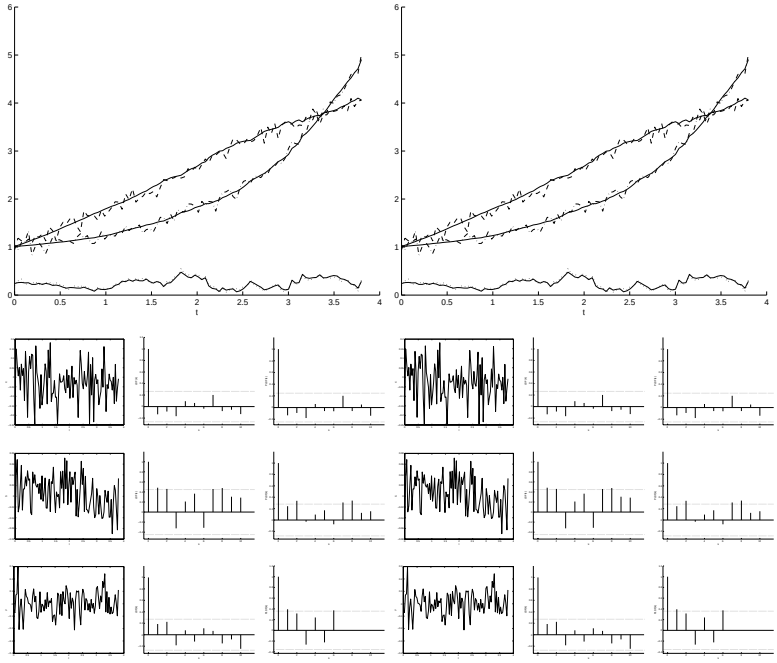


Figure E.5. Cross-validation residual analysis results for the model in (E.42) and (E.40) with parameters in Table E.3 using the data from batch no. 2 (Figure E.2b). Top left: One-step-ahead prediction comparison (solid lines: Predicted values); top right: Pure simulation comparison (solid lines: Simulated values); bottom left: One-step-ahead prediction residuals, LDF and PLDF for y_1, y_2 and y_3; bottom right: Pure simulation residuals, LDF and PLDF for y_1, y_2 and y_3.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0148E+00	1.0813E-02	93.8515	Yes
S_0	2.4127E-01	9.4924E-03	25.4177	Yes
V_0	1.0072E+00	8.7723E-03	114.8168	Yes
$\mu_{\max}$	1.0305E+00	1.7254E-02	59.7225	Yes
K_1	3.7929E-02	4.1638E-03	9.1092	Yes
K_2	5.4211E-01	2.4949E-02	21.7286	Yes
σ_{11}	2.3250E-10	2.1044E-07	0.0011	No
σ_{22}	1.4486E-07	7.9348E-05	0.0018	No
σ_{33}	3.2842E-12	3.6604E-09	0.0009	No
S_{11}	7.4828E-03	1.0114E-03	7.3982	Yes
S_{22}	1.0433E-03	1.4331E-04	7.2804	Yes
S_{33}	1.1359E-02	1.6028E-03	7.0867	Yes

Table E.3. Estimation results. Model in (E.42) and (E.40) - data from Figure E.2a.

Model falsification or unfalsification

Moving to the model falsification or unfalsification step, the re-formulated model is thus unfalsified for its intended purpose with respect to the available information, and the model development procedure can now be terminated, but, since the intended purpose of the model is to use it for an open-loop application, the parameters should ideally be re-calibrated at this point[2] with an estimation method that emphasizes the pure simulation capabilities of the model. However, this is outside the scope of the present paper.

E.3.2 Case 2: Partial state information

To illustrate that the proposed modelling cycle can also be successfully applied when only a subset of the state variables can be measured, the previous example is repeated with the assumption that only the biomass and the volume can be measured. The available sets of experimental data for this partial state information case are shown in Figure E.6. Otherwise, the same assumptions apply with respect to the intended purpose of the model and the availability of an initial model structure, where the biomass growth rate is unknown.

E.3.2.1 First modelling cycle iteration

Model formulation

The first iteration of the modelling cycle again starts with the model formulation step, where it is assumed that the model maker has been able to set up an initial model structure corresponding to (E.35)-(E.37), which is translated into a stochastic state space model with the following system equation:

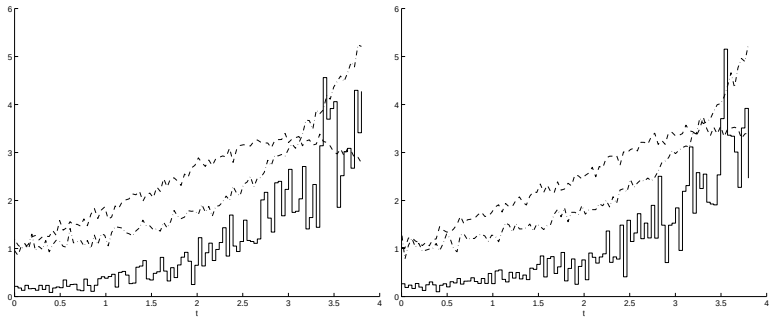
$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -\frac{\mu X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt \; + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \tag{E.43}$$

and the following modified measurement equation:

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}_k = \begin{pmatrix} X \\ V \end{pmatrix}_k + \; \boldsymbol{e}_k \;, \quad \boldsymbol{e}_k \in N(\boldsymbol{0}, \boldsymbol{S}) \;, \quad \boldsymbol{S} = \begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix} \tag{E.44}$$

where a constant biomass growth rate μ has once again been assumed, because the true structure of $\mu(S)$, which is given in (E.38), is unknown.

[2] Inspection of the t-scores for marginal tests for insignificance (Table E.3) suggest that, on a 5% level, there are no significant parameters in the diffusion term, which is confirmed by a test for simultaneous insignificance based on Wald's W-statistic.



(a) Batch no. 1. (b) Batch no. 2.

Figure E.6. The two batch data sets available for case 2. Solid staircase: Feed flow rate F; dashed lines: Biomass measurements y_1 (with $N(0, 0.01)$ noise); dash-dotted lines: Volume measurements y_2 (with $N(0, 0.01)$ noise)).

Parameter estimation

Estimating the unknown parameters of the model in (E.43)-(E.44) using the data from batch no. 1 (Figure E.6a) gives the results shown in Table E.4.

Residual analysis

Evaluating the quality of the resulting model, the cross-validation residual analysis results in Figure E.7 show that the model does a poor job in pure simulation, whereas its one-step-ahead prediction capabilities are quite good.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	9.6230E-01	1.2996E-02	74.0451	Yes
V_0	1.0272E+00	2.1417E-02	47.9641	Yes
μ	6.8730E-01	2.1875E-02	31.4198	Yes
σ_{11}	1.8846E-01	3.9179E-02	4.8104	Yes
σ_{22}	8.7290E-03	1.8577E-03	4.6989	Yes
σ_{33}	1.7391E-02	1.5107E-02	1.1512	No
S_{11}	6.7225E-03	1.0795E-03	6.2273	Yes
S_{22}	1.1078E-02	1.5137E-03	7.3184	Yes

Table E.4. Estimation results. Model in (E.43)-(E.44) - data from Figure E.6a.

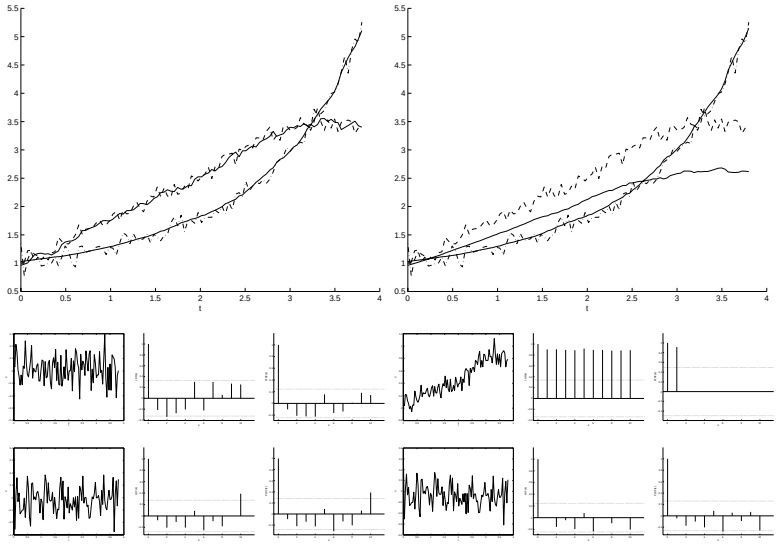


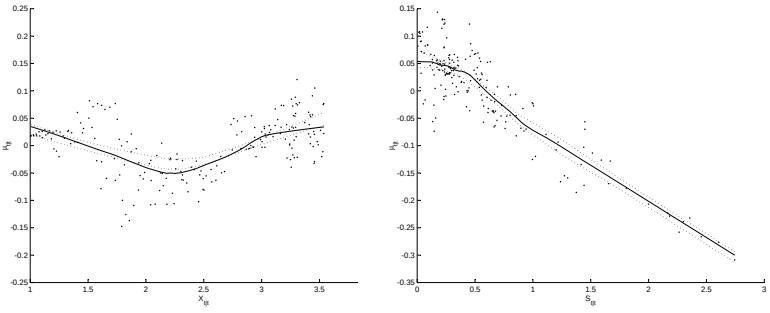
Figure E.7. Cross-validation residual analysis results for the model in (E.43)-(E.44) with parameters in Table E.4 using the data from batch no. 2 (Figure E.6b). Top left: One-step-ahead prediction comparison (solid lines: Predicted values); top right: Pure simulation comparison (solid lines: Simulated values); bottom left: One-step-ahead prediction residuals, LDF and PLDF for y_1 and y_2; bottom right: Pure simulation residuals, LDF and PLDF for y_1 and y_2.

Model falsification or unfalsification

Again the model is falsified for its intended purpose by the poor pure simulation capabilities, and the modelling cycle must therefore be repeated by reformulating the model, once its deficiencies have been pinpointed.

Statistical tests

Table E.4 also includes t-scores for performing marginal tests for insignificance of the individual parameters, and, as in the full state information case, these show that, on a 5% level, only σ_{33} is insignificant, whereas the other parameters of the diffusion term are both significant. This indicates that the first two elements of the drift term may be incorrect, and hence that μ is a possible suspect for being deficient. To confirm this suspicion the model is first re-



(a) $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$ (b) $\hat{\mu}_{k|k}$ vs. $\hat{S}_{k|k}$

Figure E.8. Partial dependence plots of $\hat{\mu}_{k|k}$ vs. $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$. Solid lines: Estimates; dotted lines: 95% bootstrap confidence intervals (1000 replicates).

formulated with μ as an additional state variable to yield the system equation:

$$d\begin{pmatrix} X \\ S \\ V \\ \mu \end{pmatrix} = \begin{pmatrix} \mu X - \frac{FX}{V} \\ -\frac{\mu X}{Y} + \frac{F(S_F - S)}{V} \\ F \\ 0 \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 & 0 \\ 0 & \sigma_{22} & 0 & 0 \\ 0 & 0 & \sigma_{33} & 0 \\ 0 & 0 & 0 & \sigma_{44} \end{bmatrix} d\boldsymbol{\omega}_t \qquad \text{(E.45)}$$

and the same measurement equation as in (E.44). The parameters of this model are then estimated using the same data set as before to give the results shown in Table E.5, and inspection of the t-scores again show that only σ_{44} is now significant on a 5% level, which in turn indicates that there is substantial variation in μ and thus confirms the suspicion that μ is deficient.

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0069E+00	2.1105E-02	47.7095	Yes
V_0	1.0250E+00	2.7800E-02	36.8687	Yes
μ_0	8.1305E-01	1.2223E-01	6.6516	Yes
σ_{11}	8.5637E-05	5.5485E-05	1.5434	No
σ_{22}	8.2654E-03	8.5005E-03	0.9723	No
σ_{33}	1.5241E-02	2.4948E-02	0.6109	No
σ_{44}	1.4751E-01	4.5181E-02	3.2648	Yes
S_{11}	7.7509E-03	1.1338E-03	6.8362	Yes
S_{22}	1.1118E-02	1.5652E-03	7.1033	Yes

Table E.5. Estimation results. Model in (E.45) and (E.44) - data from Figure E.6a.

Nonparametric modelling

The structural origin of the deficiency can again be uncovered by using the re-formulated model in (E.45) and (E.44) and the parameter estimates in Table E.5 to obtain state estimates $\hat{X}_{k|k}$, $\hat{S}_{k|k}$, $\hat{V}_{k|k}$, $\hat{\mu}_{k|k}$, $k = 0, \ldots, N$, and by fitting an additive model to reveal the true structure of the function describing μ. Assuming again that μ does not depend on V and F, the partial dependence plots shown in Figure E.8 are obtained. In this case there seems to be a dependence between $\hat{\mu}_{k|k}$ and both $\hat{X}_{k|k}$ and $\hat{S}_{k|k}$. However, since the dependence on $\hat{S}_{k|k}$ is much stronger than the dependence on $\hat{X}_{k|k}$, this again suggests to replace the assumption of constant μ with an assumption of μ being a function of S when the model is re-formulated for the next iteration.

E.3.2.2 Second modelling cycle iteration

Model re-formulation

Although less obvious, the functional relation revealed in the partial dependence plot between $\hat{\mu}_{k|k}$ and $\hat{S}_{k|k}$ in Figure E.8, is again an indication to a skilled model maker that the growth rate of biomass can be appropriately described with Monod kinetics and substrate inhibition, which allows the model to be re-formulated to yield the following system equation:

$$d\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ -\frac{\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} dt + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} d\boldsymbol{\omega}_t \tag{E.46}$$

where $\mu(S)$ is given by the true structure in (E.38), while the measurement equation remains unchanged and is therefore the same as in (E.44).

Parameter estimation

Estimating the unknown parameters of the re-formulated model using the same data set as before gives the results shown in Table E.6.

Residual analysis

Examining the cross-validation residual analysis results shown in Figure E.9, there still seems to be some non-random variation left in the cross-validation data set that is not explained by the model. This may be attributed to the fact that the data set used for parameter estimation and the cross-validation data set cover different ranges of state space, which, because only partial state information is available, the model is more sensitive to in this case.

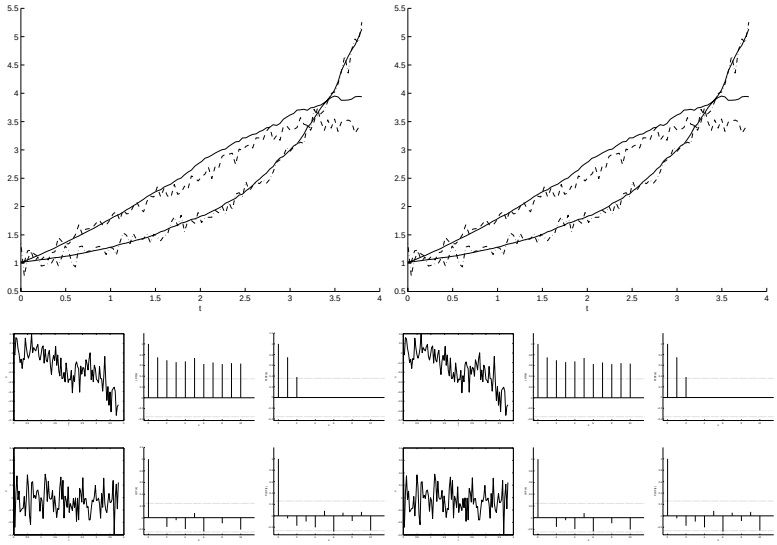


Figure E.9. Cross-validation residual analysis results for the model in (E.46) and (E.44) with parameters in Table E.6 using the data from batch no. 2 (Figure E.6b). Top left: One-step-ahead prediction comparison (solid lines: Predicted values); top right: Pure simulation comparison (solid lines: Simulated values); bottom left: One-step-ahead prediction residuals, LDF and PLDF for y_1 and y_2; bottom right: Pure simulation residuals, LDF and PLDF for y_1 and y_2.

Model falsification or unfalsification

In principle, although the results obtained with the re-formulated model are much better than those obtained with the initial model, the re-formulated

Parameter	Estimate	Standard deviation	t-score	Significant?
X_0	1.0137E+00	1.6790E-02	60.3759	Yes
V_0	1.0118E+00	1.1571E-02	87.4443	Yes
$\mu_{\max}$	1.0679E+00	1.4353E-01	7.4405	Yes
K_1	4.1664E-02	3.2800E-02	1.2702	No
K_2	6.3372E-01	1.8116E-01	3.4980	Yes
σ_{11}	6.8577E-11	2.2270E-08	0.0031	No
σ_{22}	7.9677E-06	1.1223E-03	0.0071	No
σ_{33}	1.4241E-07	2.6577E-05	0.0054	No
S_{11}	7.4094E-03	1.0986E-03	6.7447	Yes
S_{22}	1.1364E-02	1.6193E-03	7.0174	Yes

Table E.6. Estimation results. Model in (E.46) and (E.44) - data from Figure E.6a.

model is thus falsified for its intended purpose, and the modelling cycle should be repeated by re-formulating the model again. However, in the context of the proposed framework, all information available in the data set used for estimation has been exhausted, because a model has been developed where the diffusion term is insignificant[3]. In other words it is not possible to pinpoint any model deficiencies directly, because these deficiencies are only revealed by the cross-validation data set and not by the data set used for estimation. Ideally, the parameters of the model should thus be re-estimated using the cross-validation data set as well before re-formulating the model, but this takes away the possibility of easily evaluating the quality of the resulting model through cross-validation, unless more data is obtained. A discussion of possible ways to resolve this issue is outside the scope of the present paper.

E.4 Discussion

The example presented in the previous section illustrates the strength of the proposed grey-box modelling framework in terms of facilitating systematic model improvement. A key feature in this regard is the ability to pinpoint and subsequently uncover the structural origin of model deficiencies by means of estimates of unknown functional relations, and another key result is that this is also possible in situations where all process variables cannot be measured.

More specifically, the full state information case demonstrates that a high quality estimate of the functional relation between the unmeasured biomass growth rate and the measured substrate concentration can easily be obtained, and the partial state information case demonstrates that a similar estimate, of lower quality, can be obtained without measuring the substrate concentration.

The lower quality of the estimate obtained in the partial state information case is due to the fact that the performance of the proposed framework is limited by the quality and amount of available experimental data, in the sense that, if the available data is insufficiently informative, e.g. due to large measurement noise, or if the available measurements render certain subsets of the state variables of the system unobservable, parameter identifiability and hence the reliability of the proposed methods for pinpointing and uncovering the structural origin of model deficiencies is affected. Experimental design and selection of appropriate measurements are therefore key issues that must also be addressed in model development, but these are outside the scope of the present paper.

The performance of the proposed grey-box modelling framework is also limited by the quality and amount of available prior information, and if there is insufficient information to establish an initial model structure, it may not be worthwhile to use this approach as opposed to a black-box modelling ap-

[3]Inspection of the t-scores for marginal tests for insignificance (Table E.6) suggest that, on a 5% level, there are no significant parameters in the diffusion term, which is confirmed by a test for simultaneous insignificance based on Wald's W-statistic.

proach. Furthermore, the model maker must be able to determine the specific phenomenon causing a pinpointed model deficiency in order to uncover its structural origin, and this may not always be possible either.

If, however, sufficient prior information and experimental data is available, the proposed framework is very powerful as a tool for systematic model improvement. In particular, it relies less on the model maker than other approaches to grey-box modelling (Bohlin and Graebe, 1995; Bohlin, 2001) and also prevents him or her from having to resort to using black-box models for filling gaps in the model. This is due to the fact that estimates of unknown functional relations can be obtained and visualized directly. The proposed framework may be seen as a grey-box model generalization of the well-developed methodologies for identification of linear black-box models (Box and Jenkins, 1976; Ljung, 1987; Söderström and Stoica, 1989). However, unlike in the linear case, where convergence is guaranteed if certain conditions of identifiability of parameters and persistency of excitation of inputs are fulfilled, no rigorous proof of convergence exists for the framework proposed here. Nevertheless, the example presented in the previous section demonstrates that the proposed framework can be used to obtain valuable information to facilitate faster model development.

E.5 Conclusion

A systematic framework for improving the quality of continuous time models of dynamic systems based on experimental data has been presented. The proposed grey-box modelling framework is based on an interplay between stochastic differential equation modelling, statistical tests and nonparametric modelling and provides features that allow model deficiencies to be pinpointed and the structural origin of these deficincies to be uncovered to improve the model. A key result in this regard is that the proposed framework can be used to obtain nonparametric estimates of unknown functional relations, which allows unknown or inappropriately modelled phenomena to be uncovered and proper parametric expressions to be inferred from the estimated functional relations.

The performance of the proposed framework has been illustrated with an example involving a dynamic model of a fed-batch bioreactor, where it has been shown how an inappropriately modelled biomass growth rate can be uncovered and a proper parametric expression inferred. A key point illustrated with this example is that reasonable estimates of functional relations involving only variables that cannot be measured directly can also be obtained.

Abbreviations

API	Application program interface
CLDF	Crossed lag dependence function
CPU	Central processing unit
CV	Cross-validation
CTSM	Continuous Time Stochastic Modelling
EKF	Extended Kalman filter
EMM	Efficient Method of Moments
GMM	Generalized Method of Moments
II	Indirect Inference
LDF	Lag dependence function
LTI	Linear time-invariant
LTV	Linear time-varying
LS	Least squares
MARS	Multivariate Adaptive Regression Splines
MART	Multiple Additive Regression Trees
MAP	Maximum a posteriori
MCMC	Markov Chain Monte Carlo
MEF	Martingale Estimating Function
ML	Maximum likelihood
MPC	Model predictive control
NL	Nonlinear
NLDF	Nonlinear lag dependence function
NLP	Nonlinear program
NLS	Nonlinear least squares
ODE	Ordinary differential equation
OE	Output error
PE	Prediction error
PED	Prediction error decomposition
PEF	Prediction-Based Estimating Function
PEFM	Prediction-Based Estimating Function with Measurement noise
PLDF	Partial lag dependence function
SACF	Sample autocorrelation function
SCCF	Sample cross-correlation function
SDAE	Stochastic differential algebraic equation
SDE	Stochastic differential equation
SPACF	Sample partial autocorrelation function
SQP	Sequential quadratic programming
SVD	Singular value decomposition
WLS	Weighted least squares

List of publications

The following is a complete list of papers, authored or co-authored by the author of this thesis, which have been published or submitted for publication:

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2001). Computer Aided Continuous Time Stochastic Process Modelling. In R. Gani and S. B. Jørgensen, editors, *European Symposium on Computer Aided Process Engineering - 11*, pages 189–194. Elsevier.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). Using Continuous Time Stochastic Modelling and Nonparametric Statistics to Improve the Quality of First Principles Models. In J. Grievink and J. van Schijndel, editors, *European Symposium on Computer Aided Process Engineering - 12*, pages 901–906. Elsevier.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). An Investigation of some Tools for Process Model Identification for Prediction. Accepted for publication in A. S. Asprey and S. Macchietto, editors, *Dynamic Model Development: Methods, Theory and Application*, Elsevier.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). Parameter Estimation in Stochastic Grey-Box Models. Submitted for publication.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). A Method for Systematic Improvement of Stochastic Grey-Box Models. Submitted for publication.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). Stochastic Grey-Box Modelling as a Tool for Improving the Quality of First Engineering Principles Models. Submitted to ADCHEM, Hong Kong, China, 2003.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). Developing Phenomena Models from Experimental Data. Submitted to ESCAPE, Lappeenranta, Finland, 2003.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). A Unified Framework for Systematic Model Improvement. Submitted to PSE, Kun Ming, China, 2003.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002). Identification of Continuous Time Models Using Discrete Time Data. Submitted to SYSID, Rotterdam, The Netherlands, 2003.

Szederkényi, G.; Kristensen, N. R.; Hangos, K. M. and Jørgensen, S. B. (2001). Nonlinear Analysis and Control of a Continuous Fermentation Process. In R. Gani and S. B. Jørgensen, editors, *European Symposium on Computer Aided Process Engineering - 11*, pages 787–792. Elsevier.

Szederkényi, G.; Kristensen, N. R.; Hangos, K. M. and Jørgensen, S. B. (2002). Nonlinear Analysis and Control of a Continuous Fermentation Process. *Computers and Chemical Engineering*, **26**(4-5), 659–670.

References

Allgöwer, F. and Zheng, A., editors (2000). *Nonlinear Model Predictive Control (Progress in Systems & Control Theory, Vol. 26)*. Birkhauser Verlag, Switzerland.

Bajpai, R. K. and Reuss, R. (1981). Evaluation of Feeding Strategies in Carbon-Regulated Secondary Metabolite Production Through Mathematical Modelling. *Biotechnology and Bioengineering*, **13**, 717–738.

Bak, J.; Madsen, H. and Nielsen, H. A. (1999). Goodness of Fit of Stochastic Differential Equations. In P. Linde and A. Holm, editors, *Symposium i Anvendt Statistik*. Copenhagen Business School, Copenhagen, Denmark.

Bard, Y. (1974). *Nonlinear Parameter Estimation*. Academic Press, New York, USA.

Bibby, B. M. and Sørensen, M. (1995). Martingale Estimating Functions for Discretely Observed Diffusion Processes. *Bernoulli*, **1**, 17–39.

Bibby, B. M. and Sørensen, M. (1996). On Estimation of Discretely Observed Diffusions: A Review. *Theory of Stochastic Processes*, **2**(18), 49–56.

Bierman, G. J. (1977). *Factorization Methods for Discrete Sequential Estimation*. Academic Press, New York, USA.

Bitmead, R. R.; Gevers, M. and Wertz, V. (1990). *Adaptive Optimal Control - The Thinking Man's GPC*. Prentice-Hall, New York, USA.

Bohlin, T. (2001). A Grey-Box Process Identification Tool: Theory and Practice. Technical Report IR-S3-REG-0103, Department of Signals, Sensors and Systems, Royal Institute of Technology, Stockholm, Sweden.

Bohlin, T. and Graebe, S. F. (1995). Issues in Nonlinear Stochastic Grey-Box Identification. *International Journal of Adaptive Control and Signal Processing*, **9**, 465–490.

Bonvin, D. (1998). Optimal Operation of Batch Reactors - A Personal View. *Journal of Process Control*, **8**(5-6), 355–368.

Box, G. E. P. and Jenkins, J. M. (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco, USA.

Brockwell, P. J. and Davis, R. A. (1991). *Time Series: Theory and Methods*. Springer-Verlag, New York, USA, second edition.

Clarke, D. W.; Mohtadi, C. and Tuffs, P. S. (1987a). Generalized Predictive Control: I. The Basic Algorithm. *Automatica*, **23**(2), 137–148.

Clarke, D. W.; Mohtadi, C. and Tuffs, P. S. (1987b). Generalized Predictive Control: II. Extensions and Interpretations. *Automatica*, **23**(2), 149–160.

Cuthrell, J. E. and Biegler, L. T. (1989). Simultaneous Optimization and Solution Methods for Batch Reactor Control Profiles. *Computers and Chemical Engineering*, **13**(1-2), 49–62.

Dennis, J. E. and Schnabel, R. B. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, USA.

Dochain, D. and Bastin, G. (1988). Adaptive Control of Fed-Batch Fermentation Processes. In M. Kümmel, editor, *Proceedings of the IFAC Symposium on Adaptive Control of Chemical Processes*, pages 109–114. Pergamon Press.

Fletcher, R. and Powell, J. D. (1974). On the Modification of LDL^T Factorizations. *Math. Comp.*, **28**, 1067–1087.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models*. Chapman & Hall, London, England.

Hastie, T. J.; Tibshirani, R. J. and Friedman, J. (2001). *The Elements of Statistical Learning - Data Mining, Inference and Prediction*. Springer-Verlag, New York, USA.

Heyde, C. C. (1997). *Quasi-Likelihood and Its Application - A General Approach to Optimal Parameter Estimation*. Springer-Verlag, New York, USA.

Hindmarsh, A. C. (1983). ODEPACK, A Systematized Collection of ODE Solvers. In R. S. Stepleman, editor, *Scientific Computing (IMACS Transactions on Scientific Computation, Vol. 1)*, pages 55–64. North-Holland, Amsterdam.

Holst, J.; Holst, U.; Madsen, H. and Melgaard, H. (1992). Validation of Grey Box Models. In L. Dugard; M. M'Saad and I. D. Landau, editors, *Selected Papers from the 4th IFAC Symposium on Adaptive Systems in Control and Signal Processing*, pages 407–414. Pergamon Press.

Huber, P. J. (1981). *Robust Statistics*. Wiley, New York, USA.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. Academic Press, New York, USA.

Kloeden, P. E. and Platen, E. (1992). *Numerical Solution of Stochastic Differential Equations*. Springer-Verlag, Berlin, Germany.

Kotz, S. and Johnson, N. L., editors (1985). *Encyclopedia of Statistical Sciences, Vol. 5*. Wiley, New York, USA.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2001). Computer Aided Continuous Time Stochastic Process Modelling. In R. Gani and S. B. Jørgensen, editors, *European Symposium on Computer Aided Process Engineering - 11*, pages 189–194. Elsevier.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002a). Using Continuous Time Stochastic Modelling and Nonparametric Statistics to Improve the Quality of First Principles Models. In J. Grievink and J. van Schijndel, editors, *European Symposium on Computer Aided Process Engineering - 12*, pages 901–906. Elsevier.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002b). Parameter Estimation in Stochastic Grey-Box Models. Submitted for publication.

Kristensen, N. R.; Madsen, H. and Jørgensen, S. B. (2002c). A Method for Systematic Improvement of Stochastic Grey-Box Models. Submitted for publication.

Kristensen, N. R.; Melgaard, H. and Madsen, H. (2002d). *CTSM 2.1 - User's Guide*. Technical University of Denmark, Lyngby, Denmark.

Kuhlmann, C.; Bogle, I. D. L. and Chalabi, Z. S. (1998). Robust Operation of Fed Batch Fermenters. *Bioprocess Engineering*, **19**, 53–59.

Lee, K. S.; Chin, I. S.; Lee, H. J. and Lee, J. H. (1999). A Model Predictive Control Technique Combined with Iterative Learning for Batch Processes. *AIChE Journal*, **45**(10), 2175–2187.

Ljung, L. (1987). *System Identification: Theory for the User*. Prentice-Hall, New York, USA.

Madsen, H. and Melgaard, H. (1991). The Mathematical and Numerical Methods Used in CTLSM. Technical Report 7, IMM, Technical University of Denmark, Lyngby, Denmark.

Martinez, E. C. and Wilson, J. A. (1998). A Hybrid Neural Network - First Principles Approach to Batch Unit Optimisation. *Computers and Chemical Engineering*, **22**, S893–S896.

Maybeck, P. S. (1982). *Stochastic Models, Estimation, and Control*. Academic Press, London, England.

Melgaard, H. and Madsen, H. (1993). CTLSM - A Program for Parameter Estimation in Stochastic Differential Equations. Technical Report 1, IMM, Technical University of Denmark, Lyngby, Denmark.

Moler, C. and van Loan, C. F. (1978). Nineteen Dubious Ways to Compute the Exponential of a Matrix. *SIAM Review*, **20**(4), 801–836.

Muske, K. R. and Rawlings, J. B. (1993). Model Predictive Control with Linear Models. *AIChE Journal*, **39**(2), 262–287.

Nielsen, H. A. and Madsen, H. (2001a). A Generalization of some Classical Time Series Tools. *Computational Statistics and Data Analysis*, **37**(1), 13–31.

Nielsen, J. N. and Madsen, H. (2001b). Applying the EKF to Stochastic Differential Equations with Level Effects. *Automatica*, **37**, 107–112.

Nielsen, J. N.; Madsen, H. and Young, P. C. (2000a). Parameter Estimation in Stochastic Differential Equations: An Overview. *Annual Reviews in Control*, **24**, 83–94.

Nielsen, J. N.; Nolsøe, K. and Madsen, H. (2000b). Estimating Functions for Discretely Observed Diffusions with Measurement Noise. In R. Smith, editor, *Proceedings of the IFAC Symposium on System Identification*, pages 1139–1144. Elsevier.

Psichogios, D. C. and Ungar, L. H. (1992). A Hybrid Neural Network - First Principles Approach to Process Modeling. *AIChE Journal*, **38**(10), 1499–1511.

Raisch, J. (2000). Complex Systems - Simple Models? In L. T. Biegler; A. Brambilla; C. Scali and G. Marchetti, editors, *Proceedings of the IFAC Symposium on Advanced Control of Chemical Processes*, pages 275–286. Elsevier.

Ruppen, D.; Benthack, C. and Bonvin, D. (1995). Optimization of Batch Reactor Operation under Parametric Uncertainty - Computational Aspects. *Journal of Process Control*, **5**(4), 235–240.

Sidje, R. B. (1998). Expokit: A Software Package for Computing Matrix Exponentials. *ACM Transactions on Mathematical Software*, **24**(1), 130–156.

Söderström, T. and Stoica, P. (1989). *System Identification*. Prentice-Hall, New York, USA.

Speelpenning, B. (1980). Compiling Fast Partial Derivatives of Functions Given by Algorithms. Technical Report UILU-ENG 80 1702, University of Illinois-Urbana, Urbana, USA.

Srinivasan, B.; Palanki, S. and Bonvin, D. (2002a). Dynamic Optimization of Batch Processes: I. Characterization of the Nominal Solution. Accepted for publication in *Computers and Chemical Engineering*.

Srinivasan, B.; Palanki, S.; Visser, E. and Bonvin, D. (2002b). Dynamic Optimization of Batch Processes: II. Role of Measurements in Handling Uncertainty. Accepted for publication in *Computers and Chemical Engineering*.

Su, H.-T.; Bhat, N.; Minderman, P. A. and McAvoy, T. J. (1993). Integrating Neural Networks with First Principles Models for Dynamic Modeling. In J. G. Balchen, editor, *Selected Papers from the 3rd IFAC Symposium on Dynamics and Control of Chemical Reactors, Distillation Columns and Batch Processes*, pages 327–332. Pergamon Press.

Sørensen, M. (1999). Prediction-Based Estimating Functions. Technical report, Department of Theoretical Statistics, University of Copenhagen, Copenhagen, Denmark.

Thornton, C. L. and Bierman, G. J. (1980). UDU^T Covariance Factorization for Kalman Filtering. In C. T. Leondes, editor, *Control and Dynamic Systems*. Academic Press, New York, USA.

Unbehauen, H. (1996). Some New Trends in Identification and Modeling of Nonlinear Dynamical Systems. *Applied Mathematics and Computation*, **78**, 279–297.

Unbehauen, H. and Rao, G. P. (1990). Continuous-Time Approaches to System Identification - A Survey. *Automatica*, **26**(1), 23–35.

Unbehauen, H. and Rao, G. P. (1998). A Review of Identification in Continuous-Time Systems. *Annual Reviews in Control*, **22**, 145–171.

van Impe, J. F. M. and Bastin, G. (1995). Optimal Adaptive Control of Fed-Batch Fermentation Processes. *Control Engineering Practice*, **3**(7), 939–954.

van Loan, C. F. (1978). Computing Integrals Involving the Matrix Exponential. *IEEE Transactions on Automatic Control*, **23**(3), 395–404.

Visser, E. (1999). *A Feedback-Based Implementation Scheme for Batch Process Optimization*. Ph.D. thesis, École Polytechnique Fédérale De Lausanne, Lausanne, Switzerland.

Young, P. C. (1981). Parameter Estimation for Continuous-Time Models - A Survey. *Automatica*, **17**(1), 23–39.

Øksendal, B. (1998). *Stochastic Differential Equations - An Introduction with Applications*. Springer-Verlag, Berlin, Germany, fifth edition.

Åström, K. J. (1970). *Introduction to Stochastic Control Theory*. Academic Press, New York, USA.